



Health management of industrial vehicles and fleet maintenance optimization : taking into account operation constraints and mission planning

Elodie Robert

► To cite this version:

Elodie Robert. Health management of industrial vehicles and fleet maintenance optimization : taking into account operation constraints and mission planning. Automatic. Université Grenoble Alpes, 2019. English. NNT : 2019GREAT098 . tel-02901480

HAL Id: tel-02901480

<https://theses.hal.science/tel-02901480>

Submitted on 17 Jul 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



THÈSE

Pour obtenir le grade de

DOCTEUR DE LA COMMUNAUTÉ UNIVERSITÉ GRENOBLE ALPES

Spécialité : AUTOMATIQUE - PRODUCTIQUE

Arrêté ministériel : 25 mai 2016

Présentée par

Elodie ROBERT

Thèse dirigée par **Christophe BERENGUER**, Professeur des Universités, Grenoble INP

préparée au sein du **Laboratoire Grenoble Images Parole Signal Automatique (GIPSA-Lab)**
dans l'**École Doctorale Electronique, Electrotechnique, Automatique, Traitement du Signal (EEATS)**

Gestion de l'état de santé de véhicules pour la maintenance de flotte : prise en compte des contraintes opérationnelles et optimisation conjointe des maintenances et des missions

Health management of industrial vehicles and fleet maintenance optimization: taking into account operation constraints and mission planning

Thèse soutenue publiquement le **19 décembre 2019**,
devant le jury composé de :

Monsieur Yannick FREIN

Professeur des Universités, Grenoble INP, Président

Madame Anne BARROS

Professeure des Universités, Centrale-Supélec, Rapporteur

Monsieur Pierre DEHOMBREUX

Professeur, Université de Mons, Rapporteur

Monsieur Van Phuc DO

Maître de Conférences, Université de Lorraine, Examineur

Monsieur Romain LESOBRE

Ingénieur recherche (Dr.), Arquus, Examineur

Monsieur Christophe BERENGUER

Professeur des Universités, Grenoble INP, Directeur de thèse

Monsieur Marc BOURGEOIS

Manager, Volvo Group, Invité

Remerciements

Cette thèse s’est déroulée en collaboration avec le laboratoire GIPSA et le Groupe Volvo (Renault Trucks). Je souhaiterais remercier l’ensemble des personnes rencontrées durant ces trois merveilleuses années.

Je tiens tout d’abord à remercier mon formidable directeur de thèse, Christophe Bérenguer, pour son soutien sans faille, sa patience et ses conseils qui ont largement contribué à la réalisation de ce travail.

Un chaleureux merci à mes mentors côté entreprise, Keomany Bouvard, Romain Lesobre et Hocéane Tedie, sans qui ce sujet n’aurait jamais vu le jour. Leurs conseils et leurs connaissances industrielles ont été un atout indispensable. Nos petites réunions mensuelles de suivi vont me manquer !

Merci à Anne Barros et Pierre Dehombreux d’avoir accepté d’être rapporteurs, pour leurs retours enrichissants et l’intérêt porté à ce travail de thèse. Merci également aux membres de mon jury, Yannick Frein, président du jury, et Van Phuc Do, pour leurs questions pertinentes.

Il y a également des remerciements spéciaux que je tiens à adresser à toutes les personnes qui ont été présentes pour moi durant ces trois années, pour leur soutien moral et les moments chaleureux que nous avons partagés. Dans une expérience de thèse, c’est fondamental et ça compte beaucoup pour moi.

Je commence par remercier la team thésard de Volvo, les fusées de l’innovation: Johana, Donatien et Francesco. Vous êtes des as ! Spéciale dédicace à Johana, mon accolyte de voyage en Islande. On trouve quand même le temps de s’amuser pendant la thèse... sauf à la fin !

Toute ma gratitude à ma très chère équipe chez Volvo, composée de Guillaume, Célestin, Wilfried, Marc et Thomas. Vous avez fait de cette expérience de thèse une expérience exceptionnelle par votre bonne humeur ! Une mention spéciale pour mon super stagiaire Pedro qui démarre sa thèse bientôt. Bon courage à toi !

Pour terminer, j’adresse toute ma reconnaissance à des personnes très chères à mon coeur, mes parents, mon frère et mon neveu, pour leur soutien et leur présence indéfectible. Merci.

Contents

List of Figures	ix
List of Tables	xiv
Table of Acronyms and Abbreviations	xvii
Extended Summary (in French)	xxv
1 General introduction	1
1.1 Maintenance management for industrial vehicles	2
1.1.1 Main contributors to the maintenance activity	4
1.1.2 How does Volvo Group build a maintenance contract?	5
1.1.3 How to define the maintenance operations schedule in the Volvo Group?	7
1.2 Towards an improvement in the service offers	8
1.2.1 Customers needs	9
1.2.2 Considering the technological and societal evolutions	10
1.2.3 Contributions to improve the offered services and their limitations	12
1.3 Thesis contribution	17
1.3.1 Industrial objectives	17
1.3.2 Methodological challenges and thesis contribution	19
1.4 Thesis outline	21
2 Reliability modelling	23
2.1 Reliability reminders	24
2.1.1 Reliability and dependability	24
2.1.2 The major features in reliability	25

2.1.3	Reliability estimation methods	28
2.2	Physical models	28
2.3	Survival models	29
2.3.1	Impact of the environment on the survival models	30
2.4	Deterioration models	31
2.4.1	Deterioration classes	32
2.4.1.1	The discrete deterioration models	32
2.4.1.2	The continuous deterioration models	33
2.4.2	Levy processes	34
2.4.2.1	The Wiener process	34
2.4.2.2	The Gamma process	34
2.4.3	Impact of the operating environment on the deterioration model . . .	36
2.5	Conclusion	37
3	Maintenance modelling	39
3.1	General information about maintenance	40
3.1.1	Different types of maintenance	41
3.1.2	Effect of the maintenance actions on the systems	43
3.1.2.1	Perfect maintenance	43
3.1.2.2	Minimum maintenance	43
3.1.2.3	Imperfect maintenance	44
3.1.3	Maintenance model	44
3.1.4	Performance evaluation and optimization	44
3.2	Maintenance and monitoring systems	46
3.2.1	Monitoring information	47
3.2.1.1	Monitoring of the entity state	47
	Perfect monitoring	47

Imperfect monitoring	47
Continuous monitoring	48
Discrete monitoring	48
3.2.1.2 Monitoring of the environment	48
3.2.1.3 Monitoring cost	49
3.2.2 Impact of the monitoring information on the decision-making process	49
3.3 Maintenance policies for mono-component systems	51
3.3.1 Maintenance policies based on the system lifetime law	51
3.3.1.1 Age replacement policy	52
3.3.1.2 Block replacement policy	54
3.3.1.3 Inspection-based policy	56
3.3.2 Policies based on the current system state	57
3.3.2.1 Condition-based maintenance	58
3.3.2.2 Predictive maintenance policy based on the remaining useful life	59
3.4 Maintenance policies for multi-component systems	61
3.4.1 Static grouping policies	62
3.4.1.1 Corrective maintenance grouping	62
3.4.1.2 Grouping of scheduled preventive maintenance operations . .	62
Age-based maintenance policy	62
Block replacement policy	63
Inspection-based policy	64
3.4.1.3 Opportunistic maintenance	65
3.4.2 Dynamic grouping	65
3.5 Conclusion	67
4 Optimization methods to jointly schedule maintenance and production	69

4.1	Scheduling problem to organize maintenance operations	70
4.2	Models to integrate maintenance and production in the same schedule	71
4.2.1	Interrelated models	72
4.2.2	Integrated models	74
4.2.2.1	Single asset	75
4.2.2.2	Multi assets	78
4.3	Rescheduling methods	82
4.4	Optimization methods	83
4.4.1	Exact methods	84
4.4.2	Approximate methods	85
4.4.3	Multi-agent systems	86
4.4.4	Game theory	87
4.5	Genetic Algorithm	87
4.5.1	Definition	88
4.5.2	Principle	89
4.5.3	Advantages and drawbacks	91
4.6	Conclusion	92
5	Joint scheduling of missions and maintenance for a vehicle: the static case	93
5.1	Problem definition	94
5.1.1	Hypotheses and constraints	94
5.1.2	Objective	95
5.2	Resolution approach description	97
5.2.1	Vehicle deterioration model	97
5.2.1.1	Health deterioration model	97
5.2.1.2	Missions impact on the deterioration	98
5.2.2	Decision criteria	103

5.2.3	Different resolution methods	106
5.2.3.1	An exact resolution method	106
5.2.3.2	A genetic algorithm based method	108
5.3	Performance of the method	115
5.3.1	Application example	115
5.3.2	Performance analysis	118
5.3.3	Sensitivity study	121
5.3.3.1	Impact of the ratio between the preventive and corrective maintenance costs	121
5.3.3.2	Effect of the variation of the block filling condition	123
5.3.3.3	Influence of the missions variance changes	124
5.3.4	Limitations of the genetic algorithm	128
5.4	Conclusion	128
6	Joint scheduling of missions and maintenance for a vehicle: the dynamic case	131
6.1	Problem statement	132
6.1.1	Hypotheses and constraints	132
6.1.2	Objective	133
6.2	Resolution approach description	135
6.2.1	Vehicle deterioration model	136
6.2.2	Decision criteria	136
6.2.2.1	Global maintenance cost estimation	136
6.2.2.2	Production gain estimation	137
6.2.3	Dynamic scheduling algorithm	139
6.2.3.1	Dynamic scheduling principle	139
6.2.3.2	Changes in the genetic algorithm functions	143
6.3	Evaluation of the dynamic scheduling method	144

6.3.1	Rescheduling opportunities: failure occurrences and deterioration information	144
6.3.1.1	Chosen decision criterion for the algorithms	145
6.3.1.2	Cost convergence analysis	145
6.3.1.3	Effect of the rescheduling condition ΔC_{min}	148
6.3.2	Rescheduling opportunities: failure occurrences, deterioration information and new available missions	150
6.3.2.1	Chosen decision criterion for the algorithms	152
6.3.2.2	Cost convergence analysis	152
6.3.2.3	Effect of the rescheduling limit condition ΔC_{min}	156
6.3.3	Rescheduling analysis for different criteria related to the operating incomes estimation	158
6.3.4	Limits of the dynamic scheduling algorithm	160
6.4	Conclusion	161
7	Joint scheduling of missions and maintenance operations for a fleet of vehicles	163
7.1	Fleet management problem	164
7.2	Problem description	165
7.2.1	Hypotheses and constraints	165
7.2.2	Objectives	166
7.3	Reminder: vehicle deterioration model	167
7.4	The static case	168
7.4.1	Resolution approach description	169
7.4.1.1	Decision criterion	169
7.4.1.2	Description of the GA difference between one vehicle and a fleet	170
7.4.2	Performance of the fleet static scheduling approach	171
7.4.2.1	Example for an homogeneous fleet of vehicles	172

7.4.2.2	Example for an heterogeneous fleet of vehicles	177
7.5	The dynamic case	181
7.5.1	Resolution approach description	181
7.5.1.1	Definition of the missions features	181
7.5.1.2	Decision criterion	182
7.5.1.3	Description of the rescheduling method for the fleet	184
7.5.2	Application example to analyse the performances of the fleet dynamic scheduling method	186
7.5.2.1	Analysis of the fleet schedule evolution on a simulated scenario	188
7.5.2.2	Performance analysis	192
7.6	Conclusion	198
8	Conclusions and perspectives	199
8.1	Conclusions	199
8.2	Perspectives	201
8.2.1	Short term perspectives	201
8.2.2	Long term perspectives	202
	Bibliography	204

List of Figures

1.1	Distribution of the freight transport modes in Europe in 2018 (%tkm) [160] .	2
1.2	Annual operating expenses for a long haul vehicle (2018)	3
1.3	Powertrain Operating Conditions (POC) defined by the Volvo Group for a specific truck model	5
1.4	Maintenance intervals for components related to a specific system installed on a specific truck model	7
1.5	Design methodology for maintenance	16
2.1	The concepts associated with dependability	24
2.2	Distribution function and probability density	26
2.3	General behaviour of the failure rate	27
2.4	Information about the entity operating state	29
2.5	Deterioration trajectory of an entity	32
2.6	Representation of a discrete deterioration model	33
2.7	Representation of a continuous deterioration model	33
2.8	Deterioration trajectories from a Gamma process whose parameters are $\alpha = 2$ and $\beta = 3$	36
3.1	The different types of maintenance	41
3.2	Balance between preventive and corrective maintenance	45
3.3	Update of the conditional reliability based on monitoring information	50
3.4	Age replacement policy and costs	52
3.5	Mean asymptotic maintenance costs for an age replacement policy	54
3.6	Block replacement policy and costs	55
3.7	Inspection-based policy	57
3.8	Condition-based maintenance policy	59

3.9	Predictive maintenance policy based on the risk	60
3.10	Grouped block replacement policy [6]	64
5.1	Framework to schedule the missions M_1 , M_2 and M_3 and the maintenance operations MO according to the health state	96
5.2	Deterioration trajectories following two Gamma processes with the same expected value E but with different variance V	97
5.3	Deterioration evolution in a block with 3 missions	99
5.4	Histogram (empirical probability distribution function) of the realizations of x_s , corresponding to the sum of the realizations of x_1 and x_2 for each Gamma distribution	101
5.5	Comparison between the real distribution of X_s , based on the realizations x_s , and the equivalent distribution (r.v. stands for random variable)	103
5.6	Principle of the replacement process: representation of $\mathbb{P}_{f_k}(D \geq iL)$	105
5.7	Example of a crossover operation between two individuals	110
5.8	Genetic algorithm principle	114
5.9	Deterioration trajectories for the missions when the variance of the Gamma process has changed	116
5.10	Computation time evolution for the dataset A	119
5.11	Analysis of the maintenance costs convergence towards the optimal value for criteria C_1 and C_2 for dataset A	120
5.12	Computation time evolution for the dataset B	120
5.13	Analysis of the maintenance costs convergence towards the optimal value for criteria C_1 and C_2 for dataset B	121
5.14	Failure distribution and probability density compared with the results from the simulation for dataset A	126
5.15	Failure distribution and probability density compared with the results from the simulation for the dataset with variance x2	126
5.16	Failure distribution and probability density compared with the results from the simulation for the dataset with variance x5	127

6.1	Framework to schedule and reschedule the missions according to the occurring events: new mission M_5 , deterioration measure, failure	134
6.2	Dynamic sequential algorithm principle	141
6.3	Decision-making process for the schedule update	142
6.4	Comparison of the maintenance cost convergence through the 1000 simulations between the static and the dynamic methods for the datasets A and variance x2147	147
6.5	Comparison of the maintenance cost convergence through the 1000 simulations between the static and the dynamic methods for dataset C	148
6.6	Comparison of the maintenance cost convergence between the static and the dynamic methods when ΔC_{min} is varying	149
6.7	Comparison of the operating incomes between the static and the dynamic methods for $\Delta C_{min} = \frac{C_0}{2}$	154
6.8	Distribution of the total number of rescheduling n_t	154
6.9	Distribution of the numbers of rescheduling by rescheduling cause (n_d , n_f , n_{md} and n_m	155
6.10	Comparison of the operating incomes between the static and dynamic methods when ΔC_{min} is varying	156
6.11	Comparison of the operating incomes between the three different criteria for the dynamic method	159
6.12	Comparison of the computation time between the three different criteria for the dynamic method	160
7.1	Deterioration trajectories for the same mission m for the 2 different configurations	168
7.2	Process to compare the fleet and the 1VS1 methods	172
7.3	Global maintenance cost gain earned when applying the fleet method for different values of \mathbb{P}_{max}	175
7.4	Distribution of ΔC_m for a fleet with 50% of the vehicles for each vehicle type	179
7.5	Distribution of ΔC_m for a fleet with 75% of the vehicles with type 1 and 25% with type 2	180
7.6	Distribution of ΔC_m for a fleet with 25% of the vehicles with type 1 and 75% with type 2	180

7.7	Final schedule Π followed by the fleet when considering the different schedule updates	190
7.8	Deterioration evolution for VI_1	190
7.9	Deterioration evolution for VI_2	191
7.10	Deterioration evolution for VI_3	191
7.11	Operating incomes convergence for the fleet and the 1VS1 methods	193
7.12	Distribution of the operating incomes difference ΔI_o between the two methods for the simulations	193
7.13	Distribution of the maintenance costs and delay costs difference denoted ΔC_m and ΔC_d between the two methods for the simulations	194
7.14	Distributions of the total number of rescheduling n_r and the numbers of rescheduling due to deterioration information, failure and new missions respectively denoted n_d , n_f and n_{nm}	195
7.15	Operating incomes convergence for the fleet and the 1VS1 methods	196
7.16	Distribution of the operating incomes difference ΔI_o between the two methods for the simulations	196
7.17	Distribution of the maintenance costs and delay costs difference denoted ΔC_m and ΔC_d between the two methods for the simulations	197
7.18	Distributions of the total number of rescheduling n_r and the numbers of rescheduling due to deterioration information, failure and new missions respectively denoted n_d , n_f and n_{nm}	197

List of Tables

5.1	Parameters values	101
5.2	Possible blocks for $n = 4$ missions	107
5.3	Dataset A	117
5.4	Dataset B	117
5.5	Parameters definition	117
5.6	Performance results	118
5.7	Effect of R_c when considering the dataset A and the criterion C_1	122
5.8	Effect of R_c when considering the dataset A and the criterion C_2	122
5.9	Effect of R_c when considering the dataset B and the criterion C_1	122
5.10	Effect of R_c when considering the dataset B and the criterion C_2	122
5.11	Effect of \mathbb{P}_{max} when considering the dataset A and the criterion C_1	123
5.12	Effect of \mathbb{P}_{max} when considering the dataset A and the criterion C_2	123
5.13	Effect of \mathbb{P}_{max} when considering the dataset B and the criterion C_1	123
5.14	Effect of \mathbb{P}_{max} when considering the dataset B and the criterion C_2	123
5.15	Datasets when increasing the variance by 2 or 5 and when decreasing the variance by 2	124
5.16	Schedules obtained with the criteria C_1 and C_2	125
5.17	Schedules obtained with the criteria C_1 and C_2	128
6.1	Dataset A	144
6.2	Datasets when increasing the variance by 2 or 5 and when decreasing the variance by 2	145
6.3	Dataset C	146
6.4	Maintenance costs and gains generated by the dynamic method with respect to the static method according to ΔC_{min}	150

6.5	Dataset D	151
6.6	Operating incomes and gains generated by the dynamic method with respect to the static method according to ΔC_{min}	157
6.7	Average value of the number of rescheduling according to their associated cause and the value of ΔC_{min}	157
7.1	Dataset E	173
7.2	Parameters definition for the GA	173
7.3	Maintenance costs results when comparing the two methods	174
7.4	Comparison of the gains between fleet and 1VS1 schedules examples	176
7.5	Dataset F	177
7.6	Maintenance costs convergence when comparing the fleet and the 1VS1 methods for different values of \mathbb{P}_{max} and different fleet composition	178
7.7	Parameters definition for the dynamic fleet GA	187
7.8	Dataset H	188
7.9	Schedule evolution after each rescheduling	189
7.10	Operating incomes and costs distribution	192
7.11	Mission assignment to vehicle	192

Table of Acronyms and Abbreviations

Chapter 1

GCW	Gross Combined Weight
MFOP	Maintenance Free Operating Period
MFOPS	Maintenance Free Operating Period Survivability
NC	Confidence level
POC	Powertrain Operating Conditions

Chapter 2

E	Expected value of the Gamma process
$f(t)$	Density of probability function
$F(t)$	distribution function
Ga	Gamma process
L	Failure deterioration threshold
$MTTF$	Mean Time To failure
$R(t)$	Reliability function
V	Variance of the Gamma process
$z(t)$	Failure or hazard rate
Z_0	Deterioration level at $t = 0$
$Z(t)$	Deterioration indicator
α	Shape parameter of the Gamma process
β	Scale parameter of the Gamma process
$\Gamma(x)$	Gamma function
$\Gamma(a.T_i, \beta)$	Gamma distribution whose shape and scale parameters are $a.T_i$ and β

Chapter 3

ABAO	As Bad As Old
AGAN	As Good As New
c	Preventive maintenance/replacement cost
c_{insp}	Cost of an inspection
\bar{C}_R	Mean maintenance cost per time unit on an infinite horizon for the maintenance policy R

$\overline{C_R^*}$	Mean optimal asymptotic maintenance cost for the maintenance policy R
$C(t)$	Global maintenance cost over a period t
C_c	Maintenance cost
C_∞	Average asymptotic maintenance cost
k	Additional cost for corrective maintenance
N_{insp}	Number of inspections
N_{I_i}	Number of inspections for component i
N_c	Number of corrective replacements
N_p	Number of preventive replacements
r	Risk threshold
$R(t Z(t_m))$	Remaining useful lifetime at t knowing that the deterioration level is $Z(t_m)$ at t_m
RUL	Remaining useful lifetime
RUL_r	Remaining useful lifetime for a component for a given risk r
T	Failure time
T_0	Replacement period
T_0^*	Optimal replacement time interval
T_{cycle}	Length of a renewal cycle
T_j	Inspection dates
x_i^*	Nominal replacement age for component i
$X_{G_j}^*$	Optimal maintenance age for an operation group G_j
Z_i^{opp}	Opportunistic deterioration threshold for component i
Z_i^{prev}	Preventive deterioration threshold for component i
Z_{prev}	Preventive deterioration threshold
$Z(t)$	Deterioration level at t
τ	Inspection interval
ω	Tolerance age for maintenance
ω_i	Tolerance age for maintenance for component i

Chapter 4

CBM	Condition-Based Maintenance
DTM	Delay time model
ES	Exhaustive search method
GA	Genetic algorithm
HHG	Heuristic Hybrid Game
m	Number of machines
MAS	Multi-Agent System
MILP	Mixed-Integer Linear Programming
MINLP	Mixed-Integer Non Linear Programming
NEH heuristic	Nawaz-Enscore-Ham heuristic
NSGA	Non-dominated Sorting Genetic algorithm
NSGA-II	Non-dominated Sorting Genetic algorithm II

P-ACO	Pareto ant colony optimization
PM	Preventive maintenance
RUL	Remaining Useful Lifetime
SA	Simulated Annealing
SCEP model	Supervisor, Customers, Environment, Producers model
TMO	Two-Machine Optimal scheduling algorithm
Δ	Maximum capacity of a machine
λ	Failure rate
μ	Repair rate

Chapter 5

a	Part in % of the least fitted individuals that will be replaced by mutations of the best ones
AGAN	As Good As New
b	Part in % of the least fitted individuals that will belong to the pop for next iteration
<i>BFD</i> heuristic	Best Fit Decreasing heuristic
C_0	Preventive maintenance cost
C_f	Corrective maintenance cost
C_i	Decision criterion $n^o i$
<i>CV</i>	Coefficient of variation to estimate the population dispersion in the genetic algorithm
D	Deterioration level
D_i	Date of the i^{th} environment change i.e. the i^{th} change of mission in a block of missions
D_t	Total duration of a block
E	Expected value of the Gamma process
ExM	Exact method
f	Density of probability of the deterioration increments for a block of missions
f_i	Gamma law defined by the parameters α_i and β_i
<i>FF</i> heuristic	First Fit heuristic
<i>FFD</i> heuristic	First Fit Decreasing heuristic
Fit_i	Fitness function associated to the decision criterion C_i
$Ga(\alpha_e, \beta_e)$	Gamma process whose shape and scale parameters are respectively α_e and β_e
GA	Genetic algorithm
i_{max}	Maximum number of iterations in the genetic algorithm
i_p	Cyclic number of iterations after which the population dispersion is evaluated
L	Deterioration limit threshold before failure
M_i	Mission i

MO	Maintenance operation
n	Number of missions to schedule
N_b	Number of blocks composing the schedule
$N_m(k)$	Number of missions in the block k
N_{pop}	Size of the population in the genetic algorithm
RSD	Relative standard deviation
t_0	Initial time
t_i	Duration of mission i
P_0	Population at the beginning of an iteration of the genetic algorithm
P_1	Population after the mutation operations
P_{cross}	Crossover probability for the genetic algorithm
P_{mut}	Mutation probability for the genetic algorithm
R_c	Ratio between the preventive and corrective maintenance costs C_0 and C_f
T_c	Computation time
V	Variance of the Gamma process
x_i	Realization of X_i
X_i	Random variable following a gamma distribution F_i
$X(t)$	Deterioration indicator for the vehicle
\mathbb{P}	Failure probabilities for the missions
$\mathbb{P}_f(k)$	Probability to have one failure in the block k
$\mathbb{P}_{f_k}(D \geq iL)$	Probability to exceed the failure threshold for the i^{th} time in the block k
\mathbb{P}_{max}	Maximum failure probability that a block cannot overstep to be feasible
α	Shape parameter of the Gamma process
α_e	Shape parameter of the equivalent Gamma process followed by the deterioration in a block of missions
β_e	Scale parameter of the equivalent Gamma process followed by the deterioration in a block of missions
α_m	Shape parameter of the Gamma process followed by the vehicle deterioration when it executes the mission m
β	Scale parameter of the Gamma process
β_m	Scale parameter of the Gamma process followed by the vehicle deterioration when it executes the mission m
ε_{max}	Maximum population dispersion threshold value
ε_{min}	Minimum population dispersion threshold value
μ	Mean value of the fitness values for the population individuals
π	Schedule
$\pi_{opt,J}^i$	Optimal schedule for dataset J when using the decision criterion C_i
$\pi_{opt,xK}^i$	Optimal schedule for the dataset with a variance multiplied by a factor K when using the decision criterion C_i
$\pi_{opt K}^i$	Optimal schedule for the dataset with a variance divided by a factor K when using the decision criterion C_i
σ	Standard deviation of the fitness values for the population individuals

Chapter 6

B_b	Block b
$c_d(i)$	Delay cost for mission i
$c_{ud}(i)$	Unitary delay cost for mission i
C_0	Preventive maintenance cost
C_f	Corrective maintenance cost
C_i	Decision criterion n ^o i
C_m	Global maintenance costs
$d(j)$	Duration of mission j
d_c	Duration of a corrective maintenance operation
d_p	Duration of a preventive maintenance operation
$d_{s_{max}}(i)$	Starting deadline for mission i
D	Deterioration level
$g_m(i)$	Raw gain earned when finishing mission i without delay
G	Global production gain
G_p	Criterion to estimate the production gain generated by the missions
n	Number of missions to schedule
n_d	Number of rescheduling due to deterioration measurements
n_f	Number of rescheduling due to failure occurrences
n_m	Number of rescheduling due to new requested missions
n_{md}	Number of rescheduling due to new missions and deterioration measurements
n_t	Total number of rescheduling
L	Failure deterioration threshold
M_i	Mission i
MO	Maintenance operation
$N(k)$	Number of failures in the block k
N_b	Number of blocks composing the schedule
N_t	Total number of missions to schedule
RUL	Remaining Useful Lifetime
t_0	Initial time when the schedule starts being executed
$t(k, i)$	Time spent in block k before starting mission i
$t_d(i)$	Estimated delay time for mission i
$t_s(i)$	Real starting date for mission i
$\mathbb{E}[N_f(b)]$	Average number of failures in the block b
$\mathbb{E}[t_s(k)]$	Expected time when the block k begins to be completed
\mathbb{P}	Failure probabilities for the missions
$\mathbb{P}_f(k, i)$	Probability to exceed the failure threshold for the i^{th} time in the block k
\mathbb{P}_{max}	Maximum failure probability that a block cannot overstep to be feasible
α_m	Shape parameter of the Gamma process followed by the vehicle deterioration when it executes the mission m
β_m	Scale parameter of the Gamma process followed by the vehicle deterioration when it executes the mission m

ΔC_{min}	Minimum gain a rescheduling must generate to be considered
π_{ds}	Final schedule obtained with the dynamic scheduling method
π_{is}	Initial schedule obtained with the static scheduling method
π_s	Final schedule obtained with the static scheduling method
π_{s_J}	Schedule obtained with the static scheduling method for dataset J

Chapter 7

a	Part in % of the least fitted individuals that will be replaced by mutations of the best ones
b	Part in % of the least fitted individuals that will belong to the pop for next iteration
B_k	Block k
BFD heuristic	Best Fit Decreasing heuristic
$c_d(m)$	Delay cost for mission m
c_l	Loading/unloading costs induced for a switch of vehicle for an on going mission
C_0	Preventive maintenance cost
C_d	Delay costs
C_f	Corrective maintenance cost
C_i	Decision criterion n ^o i
C_l	Loading/unloading cost
C_m	Maintenance cost
C_m^{fleet}	Decision criterion to estimate the global maintenance costs for the fleet when applying the fleet-based scheduling algorithm
C_m^{1VS1}	Decision criterion to estimate the global maintenance costs for the fleet when applying the 1VS1 scheduling method
$d(m)$	Duration of mission m
d_c	Duration of a corrective maintenance operation
$d_{max}(m)$	Deadline of mission m
d_p	Duration of a preventive maintenance operation
FF heuristic	First Fit heuristic
FFD heuristic	First Fit Decreasing heuristic
$g_m(m)$	Raw gain earned when finishing mission m without delay
$Ga(\alpha_m, \beta_m)$	Gamma process followed by the deterioration when executing the mission m whose shape and scale parameters are respectively α_m and β_m
G_p	Sum of the raw gains earned when the missions are completed
GA	Genetic algorithm
i_{max}	Maximum number of iterations for the GA (used to stop the algorithm if the stopping criterion based on I_{max} is never reached)
i_p	Cyclic number of iterations after which the population dispersion is evaluated
I_o	Operating incomes

I_{max}	Fixed number of iterations during which if there is no change in the GA population best fitness, the GA stops
L	Failure deterioration threshold
n	Number of missions to complete
n_d	Number of rescheduling due to deterioration measurements
n_f	Number of rescheduling due to failure occurrences
n_m	Number of rescheduling due to new requested missions
n_{md}	Number of rescheduling due to new missions and deterioration measurements
n_r	Total number of rescheduling
n_s	Number of scenarios
N_b	Total number of blocks for the fleet schedule
$N_b(v)$	Number of blocks composing the schedule of vehicle v
N_{pop}	Size of the population in the genetic algorithm
N_S	Total number of scenarios
N_v	Number of vehicles in the fleet
N_{v_d}	Number of vehicle configurations in the fleet
p_{mut}	Mutation probability for each gene of the individual i.e. each mission of the schedule
P_{cross}	Crossover probability for the genetic algorithm
P_{mut}	Probability to mute each individual in the genetic algorithm
PEV	Plug electric vehicle
$t(m)$	Time when mission m ends
t_0	Initial time
$t_d(m)$	Delay time for mission m
$T_{CM}(m)$	Time spent in corrective maintenance from the first mission belonging to the block containing mission m until mission m , mission m included
$T_M(m)$	Time spent in maintenance before the beginning of the block containing mission m
$T_{in_m}(m)$	Time spent in mission until mission m including the duration of mission m
V_v	Vehicle v
VI	Vehicle
$\mathbb{E}[N_f(b)]$	Average number of failures in the block b
$\mathbb{E}[N_f(b j \leq m)]$	Average number of failures during all the missions j of the block b before mission m until mission m included
\mathbb{P}	Failure probabilities for the missions
$\mathbb{P}_f(v, k)$	Probability to have a failure in the block k belonging to the schedule of the vehicle v
\mathbb{P}_{max}	Maximum failure probability that a block cannot overstep to be feasible
α	Shape parameter of the Gamma process
α_m	Shape parameter of the Gamma process followed by the vehicle deterioration when it executes the mission m
α_{m_i}	Shape parameter of the Gamma process followed by the vehicle deterioration when it executes the mission m knowing that the vehicle has the configuration i

β	Scale parameter of the Gamma process
β_m	Scale parameter of the Gamma process followed by the vehicle deterioration when it executes the mission m
β_{m_i}	Scale parameter of the Gamma process followed by the vehicle deterioration when it executes the mission m knowing that the vehicle has the configuration i
ΔC_d	Difference between the delay costs for the schedule obtained with the fleet method and the maintenance costs for the schedule obtained with the 1VS1 method
ΔC_m	Difference between the maintenance costs for the schedule obtained with the fleet method and the maintenance costs for the schedule obtained with the 1VS1 method
ΔC_{min}	Minimum gain a rescheduling must generate to be considered
ΔI_o	Difference between the operating incomes for the schedule obtained with the fleet method and the maintenance costs for the schedule obtained with the 1VS1 method
ε_{max}	Maximum population dispersion threshold value
ε_{min}	Minimum population dispersion threshold value
π_v	Schedule for the vehicle v
Π	Schedule for the fleet
Π_{d_i}	Initial schedule for the fleet

Extended Summary (in French)

En 2018, les transporteurs utilisent majoritairement les routes pour livrer leurs marchandises en Europe. Cela vient du fait que ce mode de transport est relativement peu cher et flexible pour livrer de grande quantité de marchandises jusqu'au domicile des clients. Les camions sont des outils permettant au client de générer des profits. Depuis des années, les transporteurs essaient de réduire au maximum les coûts d'opération des véhicules en les utilisant au maximum de leur capacité afin de maximiser ces profits. Ils travaillent donc constamment à flux tendu. Pour ce faire, il est nécessaire de gérer les revenus d'exploitation globaux des flottes de chaque client. La maintenance est un levier important dans les dépenses opérationnelles. Elle ne peut être négligée dans la mesure où elle assure le fonctionnement du véhicule mais permet également d'améliorer la disponibilité du véhicule si elle est planifiée à des moments opportuns dépendant de l'usure et de l'usage du véhicule.

Les clients sont intéressés par une solution de transport complète incluant des offres de services, adaptées à leur usage, permettant de gérer au mieux leur flotte de camions. Ils veulent avoir accès à des services flexibles qui s'adaptent à leur stratégie, éviter les arrêts non planifiés de leurs véhicules pour limiter les coûts additionnels dus à l'immobilisation, et être en mesure de livrer les marchandises en temps et en heure en respectant les contraintes de livraison. En effet, un retard de livraison peut engendrer un coût de pénalité conséquent sur les revenus d'exploitation.

De plus, le développement des nouvelles technologies mène à un changement graduel du modèle économique du transport routier et trois nouvelles tendances émergent et contribuent à modifier la conception des solutions de transport mais également la façon de les utiliser. Ces trois axes d'évolution sont la digitalisation, l'électromobilité et le véhicule autonome.

Les différents éléments évoqués précédemment introduisent les principaux problèmes liés à la gestion des opérations de maintenance et des livraisons pour une flotte de camions.

Afin de considérer l'ensemble de ces contraintes et les besoins clients, ce travail de thèse a pour objectif de développer des méthodes permettant d'optimiser conjointement le planning des opérations de maintenance et la gestion de flotte de véhicules. C'est un double avantage pour le client car les opérations de maintenance seront planifiées sans impacter la disponibilité opérationnelle des véhicules et la productivité est améliorée en utilisant au mieux la capacité de la flotte pour distribuer les missions entre les différents véhicules.

La méthodologie mise en place pour résoudre le problème est composée de trois étapes avec un niveau de complexité graduelle. Chaque étape permet de développer une contribution à part entière tout en définissant le socle pour le développement de l'étape suivante.

La première étape est celle de la planification conjointe des maintenances et des missions en considérant l'évolution de la dégradation du véhicule ainsi que les changements de conditions d'usage opérationnel. La stratégie d'optimisation pour obtenir le planning conjoint

maintenance/missions repose sur le groupement des missions par blocs séparés par des actions de maintenance. Ces blocs sont obtenus de sorte que la probabilité d'avoir une panne ne dépasse pas un certain seuil et afin de limiter les coûts de maintenance.

La seconde étape considère l'intégration des informations temps réel sur l'état de santé du véhicule, les occurrences de panne et les modifications des ordres de livraison afin de rendre le planning conjoint dynamique.

Enfin, la dernière étape consiste à intégrer la dimension flotte de véhicules pour prendre les décisions de maintenance et de livraison, non plus au niveau du véhicule, mais au niveau de la flotte, afin d'utiliser au maximum les capacités offertes par la flotte.

Le travail de thèse se divise en 2 grandes parties. La première se compose des chapitres 2, 3 et 4. Il s'agit d'une étude bibliographique complète permettant de situer le problème selon trois axes et de répertorier les stratégies existantes pour le résoudre. Le chapitre 2 détaille les différentes façons de modéliser la fiabilité d'un système et se focalise sur les modèles de dégradations, notamment le processus stochastique Gamma, utilisés dans les chapitres sur la méthodologie de planification. Le chapitre 3 permet de répertorier les politiques de maintenance évoquées dans la littérature et d'évaluer la plus à même d'être adaptée pour résoudre le problème de planification conjointe maintenance/missions. La suite du manuscrit montre que nous avons choisi de nous orienter vers une politique de maintenance prévisionnelle, plus communément appelé maintenance prédictive, basée sur la durée de vie résiduelle afin de prendre des décisions sur le meilleur moment pour effectuer la maintenance. Enfin, le chapitre 4 décrit les méthodes d'optimisation utilisées dans la littérature pour résoudre les problèmes de planification conjointe maintenance et production. Il montre une grande diversité dans la description des problèmes et dans les hypothèses considérées. Les approches existantes ne permettent pas de traiter le problème de planification conjointe maintenance/missions pour des véhicules se détériorant au cours du temps de manière stochastique. Cette incertitude sur le processus de dégradation n'a pas encore été intégrée au problème de planification conjointe pour une flotte de véhicules. La seconde partie est composée des différentes contributions permettant d'atteindre l'objectif final : planifier de façon dynamique et simultanée les opérations de maintenance et les missions pour une flotte de véhicules.

Chapitre 5 : Planification conjointe des missions et des opérations de maintenance : le cas statique

Ce chapitre propose une méthode de planification conjointe maintenance/missions pour un seul véhicule dans le cas statique. On suppose que la liste complète de missions à effectuer par ce véhicule est disponible au début de la planification et qu'aucun événement ne peut modifier le planning obtenu.

Dans un premier temps, les hypothèses considérées ainsi que le problème sont décrits. La méthode de planification statique est ensuite présentée et évaluée à l'aide d'un exemple d'application. Ses performances sont également comparées à celles obtenues avec une méthode exacte. Pour terminer, une étude de sensibilité est réalisée afin d'évaluer l'impact des paramètres nécessaires à l'application de la méthode de planification statique.

5.1 Définition du problème

5.1.1 Hypothèses et contraintes

Le véhicule se détériore au cours du temps en fonction de son activité, qui dépend des conditions d'usage décrites par les missions. En effet, le véhicule effectue des missions ayant différentes sévérités d'usage, caractérisées par des durées et des paramètres liés à l'environnement différents, comme l'état des routes ou la topographie. L'usage du véhicule doit donc être modélisé pour intégrer les différents niveaux de sévérité des missions.

Nous considérons que toutes les opérations de maintenance restaurent l'état de santé du véhicule à un état comme neuf. Les opérations préventives planifiées ont un coût associé égal à C_0 alors que les opérations correctives dues à une panne ont un coût égal à C_f .

5.1.2 Objectif

L'objectif est d'optimiser le planning du véhicule en utilisant les informations de dégradation décrites par les caractéristiques des missions ainsi que le modèle de dégradation du véhicule. Le processus de décision est basé sur des estimations de probabilités de panne. Il est à noter qu'aucune mesure réelle de la dégradation n'est disponible dans ce cas.

Les opérations de maintenance préventives doivent être planifiées afin d'éviter les arrêts immobilisant dus à des pannes, de maximiser la disponibilité du véhicule et de ne pas perturber le déroulement des missions. Pour que ces conditions soient remplies, le planning final est défini en regroupant les missions en différents blocs, chaque bloc étant séparé du suivant par une opération de maintenance préventive. L'optimisation du planning est basée sur le coût de maintenance, composé des coûts préventif et correctif. Le coût de maintenance correctif est associé au risque de panne dans chaque bloc de missions. Les modèles de maintenance et de missions permettent de définir à la fois, les meilleurs intervalles de temps pour terminer les blocs de missions et effectuer les opérations de maintenance, ainsi que la meilleure manière possible de remplir les blocs de missions pour minimiser le nombre de maintenances préventives.

5.2 Description de la méthode de résolution

5.2.1 Modèle de dégradation du véhicule

5.2.1.1 Modèle de dégradation de l'état de santé

Le véhicule est considéré comme un système mono-composant dont l'état de santé est décrit à l'aide d'un indicateur global de dégradation. Etant donné que la majorité des composants d'un camion sont mécaniques et sujet à une usure graduelle, nous avons choisi d'utiliser un modèle de dégradation continue, tel que le processus Gamma. Lorsque le véhicule a un niveau de dégradation cumulé supérieur à un certain seuil, une panne se produit. Une estimation de la probabilité de panne est donc utilisée afin de prendre la décision d'envoyer le véhicule sur une autre mission après celle en cours, ou de l'envoyer à l'atelier pour effectuer une maintenance.

5.2.1.2 Impact des missions sur le processus de dégradation

Il existe un lien entre la mission à effectuer et l'évolution de la dégradation du véhicule.

Le véhicule évolue dans un environnement dynamique, ce qui influence sa dégradation. Ces variations d'environnement proviennent des changements de conditions opérationnelles du véhicule au cours des différentes missions effectuées durant sa vie.

En groupant les missions en blocs, les conditions d'usage du véhicule changent d'une mission à l'autre dans le même bloc. Il est possible de modéliser ces changements afin d'évaluer la probabilité d'avoir une panne après un certain nombre de mission. Mais le calcul devient de plus en plus complexe au fur et à mesure que le nombre de missions augmente. Pour éviter cela, nous avons choisi de modéliser l'évolution de la dégradation du véhicule dans chaque bloc à l'aide d'un modèle équivalent considérant les caractéristiques de chaque mission présente dans le bloc (processus Gamma équivalent).

5.2.2 Critère de décision

Le critère d'optimisation choisi est basé sur l'évaluation du coût de maintenance pour le planning. Ce coût est décomposé en deux parties : le coût préventif et le coût correctif. Le coût préventif correspond à la somme des opérations de maintenance effectuées à la fin de chaque bloc de missions. Le coût correctif correspond au coût relatif aux pannes pouvant se produire dans les blocs de missions.

La façon de remplir les blocs avec des missions est directement liée à l'évolution de la dégradation du véhicule et l'estimation de la probabilité d'avoir une panne dans les blocs.

Ce critère peut également considérer soit une panne, soit plusieurs pannes par bloc de missions.

5.2.3 Différentes méthodes de résolution

5.2.3.1 Méthode exacte

La méthode exacte permet de générer tous les plannings possibles respectant la condition de remplissage des blocs, notée \mathbb{P}_{max} . Seul un bloc pour lequel la probabilité de panne est inférieure à \mathbb{P}_{max} est considéré comme faisable. Il s'agit d'une méthode de recherche exhaustive.

Cependant, comme le problème de planification conjointe maintenance/missions est un problème NP-complet, on atteint l'explosion combinatoire assez rapidement lorsque le nombre de missions à planifier augmente. Le temps de calcul devient donc conséquent, ce qui montre l'intérêt d'utiliser une méthode approchée afin de trouver un planning convenable, même si sous-optimal, dans un délai raisonnable.

De plus, aucune méthode standard de planification ne peut être utilisée étant donné que la probabilité de panne pour chaque bloc change en fonction des missions placées dans ce bloc.

5.2.3.2 Méthode basée sur un algorithme génétique

La méthode de planification statique développée est basée sur un algorithme génétique. La première étape de l'algorithme consiste à définir une population initiale d'individus. Un individu correspond à un planning composé de blocs de missions séparés par une opération de maintenance. Ensuite, les individus sont évalués et classés à l'aide de la fonction objectif. Tant que le critère d'arrêt n'est pas satisfait, des opérateurs génétiques (crossover et mutation) sont appliquées sur la population pour obtenir de meilleurs individus.

L'ensemble des étapes de l'algorithme génétique est également décrite dans cette partie.

5.3 Performances de la méthode au travers d'exemples d'application

Un ensemble de 6 missions à planifier est considéré dans les exemples.

- Analyses des performances : La comparaison entre la méthode de planification basée sur l'algorithme génétique et la méthode exacte permet d'attester de la convergence de l'algorithme génétique vers le même planning que celui obtenu avec la méthode exacte. L'erreur de coût de maintenance entre les résultats de simulations et le coût estimé avec l'algorithme génétique sont de l'ordre de moins de 1%.

De plus, l'utilisation de l'algorithme génétique permet de réduire le temps de calcul nécessaire à obtenir le planning conjoint maintenance/ missions, entre 7% et 70% de gain de temps en fonction des critères de décisions considérés.

- Etude de sensibilité : Trois études de sensibilité sont menées pour trois paramètres :
 - Le ratio entre les coûts préventif et correctif : $R_c = \frac{C_0}{C_f}$
 - L'effet de la variation de la condition de remplissage des blocs \mathbb{P}_{max}
 - L'influence des changements de variance pour les processus de dégradations des missions.

Lorsque la valeur de R_c se rapproche de 0,1, cela signifie qu'une opération de maintenance corrective est dix fois plus chère qu'une opération de maintenance préventive. Grouper les missions en blocs augmente alors le risque de panne pour chaque bloc et mène à une augmentation du coût de maintenance estimé. Le but recherché étant d'éviter les pannes et de minimiser le coût de maintenance, il est plus pertinent d'avoir un planning composé de beaucoup de blocs avec peu de missions dans chacun d'eux. Au contraire, lorsque R_c se rapproche de 1, les opérations de maintenance préventive et corrective ont le même coût donc grouper les missions dans un petit nombre de blocs devient moins cher.

Lorsque la valeur de la condition de remplissage des blocs \mathbb{P}_{max} augmente, on peut donc ajouter davantage de missions dans un même bloc car la probabilité de panne admissible est plus grande. Cela mène donc à réduire le nombre de blocs composant le planning et réduit également le coût de maintenance associé au planning.

Lorsque la variance de la dégradation augmente, l'incertitude sur une potentielle occurrence de panne augmente donc la probabilité d'avoir une panne augmentée. Il devient donc plus difficile de grouper les missions en blocs sans dépasser la condition de remplissage des blocs \mathbb{P}_{max} .

Ce chapitre permet de montrer le potentiel offert par une méthode d'optimisation basée sur un algorithme génétique pour planifier de façon conjointe les maintenances et les missions pour un véhicule.

En revanche, nous n'avons étudié pour l'instant que le cas statique. La prochaine étape serait de considérer les informations temps réel sur l'état de santé du véhicule, les occurrences de pannes et les nouvelles missions à planifier pour mettre à jour le planning initial et ainsi obtenir une méthode de planification dynamique.

De même, nous ne considérons que le coût de maintenance comme critère de décision. Ajouter les coûts associés aux livraisons, notamment dus aux pénalités de retard si les livraisons ne sont pas faites à l'heure, serait judicieux. En effet, ils ont également un impact sur la façon de définir le planning conjoint maintenance/ missions optimal.

Le chapitre 6 considère ces différents éléments afin de proposer une approche pour planifier de façon conjointe les maintenances et les missions pour un véhicule dans le cas dynamique.

Chapitre 6 : Planification conjointe des missions et des opérations de maintenance : le cas dynamique

Ce chapitre propose une méthodologie dynamique de planification des missions et des opérations de maintenance pour un camion en considérant différents événements pouvant se produire pendant la vie du véhicule. Ces événements sont vus comme des opportunités de mise à jour du planning initial afin de s'adapter à l'évolution de l'usage et des nouvelles contraintes opérationnelles. Les trois types d'informations prises en compte sont les occurrences de panne, les mesures de dégradations et les nouvelles missions.

Tout d'abord, nous introduisons le problème de planification dynamique ainsi que les hypothèses considérées pour pouvoir ensuite présenter la méthode utilisée. Des exemples d'application permettent d'illustrer les résultats obtenus.

6.1 Description du problème

6.1.1 Hypothèses et contraintes

Les hypothèses considérées sont similaires à celles du chapitre 5. On suppose qu'un indicateur global de dégradation permet de suivre l'évolution de l'état de santé du véhicule au cours du temps, en fonction des variations d'usage dues aux différentes missions. La politique de maintenance appliquée est également la même que celle utilisée dans le cas statique.

En revanche, l'ensemble des missions à effectuer n'est plus connu dès le début de la planification. Cet ensemble peut évoluer au cours du temps avec l'ajout de nouvelles missions à compléter. Nous supposons également qu'il est impossible d'arrêter une mission en cours

pour mettre à jour le planning en intégrant les nouvelles missions disponibles.

Dans le cas où une panne se produit, le planning est immédiatement mis à jour pour intégrer cette information. Au contraire, si une mesure de l'état de santé est disponible, une replanification n'est pas obligatoire. Cela dépend de la criticité de l'information i.e. de son impact sur le planning en cours.

6.1.2 Objectif

L'objectif est de construire un planning conjoint missions/ maintenance adaptatif en utilisant les informations provenant des caractéristiques des missions et du modèle de dégradation du véhicule mais aussi les informations temps réel pour optimiser la stratégie de maintenance prévisionnelle (prédictive). Les différents événements temps réel seraient utilisés afin de mettre à jour de façon séquentielle le planning.

Les principaux enjeux consistent à réorganiser les missions quand de nouvelles missions doivent être ajoutées et à évaluer si une information temps réel est suffisamment significative pour déclencher une replanification.

6.2 Description de la méthode de résolution

6.2.1. Modèle de dégradation du véhicule

Le modèle de dégradation du véhicule est similaire à celui décrit dans le cas statique. Le véhicule est considéré comme un système mono-composant dont la dégradation suit un processus continu et stochastique. Nous avons choisi le processus Gamma pour modéliser cette dégradation.

Ce modèle de dégradation permet de prendre les décisions pour définir les blocs de missions en planifiant les opérations de maintenance au bon moment. Il est donc possible d'estimer la probabilité d'avoir une panne dans les blocs mais aussi la durée de vie résiduelle du véhicule étant donné que des informations sur son état de santé sont disponibles. Nous supposons que l'état de santé est disponible à la fin de certaines missions. Ces estimations permettront de choisir quand planifier les maintenances, quand déployer le véhicule sur des missions et comment mettre à jour le planning pour considérer l'état de santé courant du véhicule.

6.2.2 Critère de décision

Dans le cas dynamique, nous nous intéressons à deux coûts différents : le gain généré par la réussite des missions et le coût de maintenance associé aux opérations préventives et correctives menées durant la vie du véhicule.

Le coût de maintenance C_m est estimé en considérant les coûts des opérations préventives faites à la fin de chaque bloc de missions et les coûts des opérations correctives estimées à partir des probabilités de panne pour les blocs de missions.

Le gain de production G_p , obtenu lorsque les missions sont terminées, est calculé à partir des gains bruts de chaque mission et d'une estimation des pénalités de retard de chaque

mission. Ces pénalités sont estimées à l'aide des dates limites avant lesquelles les missions doivent avoir commencé et d'une estimation de la date de début de la mission dans le planning.

Le critère choisi correspond au revenu d'exploitation pour le véhicule ($G_p - C_m$).

6.2.3 Algorithme de planification dynamique

6.2.3.1 Principe de l'algorithme

Le planning initial est généré à l'aide de l'algorithme de planification statique décrit dans le chapitre 5. Les missions disponibles au départ sont planifiées ainsi que les opérations de maintenance préventives pour optimiser le critère de décisions.

Pendant l'exécution du planning initial, des replanifications peuvent être faites si des événements se produisent, tels que des occurrences de panne, des mesures de dégradation et des nouvelles missions requises. Les différents événements temps réel déclenchent de potentielles mises à jour du planning :

- Une panne se produit pendant une mission : Une maintenance corrective est faite sur le véhicule et une replanification des missions restantes est appliquée. La première mission à replanifier est celle en cours car on ne peut pas arrêter une mission en cours.
- Une nouvelle mission est demandée : Elle est ajoutée à la liste des missions à faire. Dès que la mission en cours est terminée, une replanification est appliquée pour intégrer cette mission dans le planning. Dans le cas où une mesure de dégradation est disponible au moment de la replanification, cette information est intégrée dans l'ordre de replanification.
- Une mesure de dégradation est disponible :
 - Si cette mission est à la fin d'un bloc, on estime le potentiel gain généré si on replanifie et on le compare au gain minimum pour autoriser une replanification ΔC_{min} . Si on ne gagne pas au minimum ΔC_{min} , on conserve le planning en cours.
 - Si cette mission se situe dans un bloc de mission, on commence par estimer la probabilité d'avoir une panne dans ce bloc. Dans le cas où cette probabilité dépasse la probabilité maximum admissible \mathbb{P}_{max} , la replanification est automatiquement appliquée. Sinon, on estime le potentiel gain généré si on replanifie et on accepte la replanification uniquement si ce gain est supérieur à ΔC_{min} .

6.2.3.2 Algorithme génétique

L'implémentation de cette méthode de planification dynamique est basé sur l'algorithme génétique développé dans le cas statique. Des adaptations sont mises en place pour prendre en compte les informations temps réel et les intégrer dans la stratégie de prise de décisions.

6.3 Evaluation de la méthodologie de planification dynamique

6.3.1 Opportunités de replanification : les occurrences de panne et les mesures de dégradation

Dans ce premier exemple, nous étudions les performances de la méthode en ne considérant que les occurrences de panne et les mesures de dégradation comme événement permettant la mise à jour du planning.

Le critère de décision utilisé pour optimiser les décisions de planification est basé uniquement sur le coût de maintenance C_m . En fonction de la liste de missions considérées et des paramètres d'usage associés aux missions, la méthodologie dynamique permet de réduire le coût de maintenance de 11 à 25% par rapport à la méthode statique n'autorisant aucune replanification.

Ces résultats mettent en évidence l'importance d'intégrer les informations de surveillance dans la stratégie de prise de décision et d'adapter le planning conjoint maintenance/ missions initial. En effet, les informations de surveillance ont une vraie valeur ajoutée pour prendre les meilleures décisions et éviter des dépenses inutiles pour des actions de maintenance.

De plus, la condition de replanification décrite par le coût minimum pour qu'une replanification soit acceptée, ΔC_{min} , permet d'éviter de modifier trop souvent le planning dans le cas où des mesures de l'état de santé du véhicule sont disponibles. Le coût de maintenance diminue quand la valeur de ΔC_{min} diminue. La replanification devient quasiment une action automatique quand une mesure de dégradation est disponible, c'est-à-dire à la fin de presque chaque mission. En effet, dans les exemples d'application, on considère que si une mesure de dégradation est disponible, elle ne peut l'être qu'à la fin d'une mission. Encourager des replanifications permet donc de réduire les coûts de maintenance. Cependant, nous ne prenons pas en compte dans cette étude les coûts associés à la replanification, tels que les coûts logistiques ou les dépenses dues à la récupération des informations de surveillance. En les considérant, on pourrait estimer la valeur optimale de ΔC_{min} permettant de rentabiliser la récupération de données de surveillance.

6.3.2 Opportunités de replanification : occurrences de panne, mesures de dégradation et nouvelles missions

Dans cette partie, nous étudions les performances de l'algorithme en considérant, en plus des occurrences de panne et des mesures de dégradation, les nouvelles missions requises comme une opportunité de replanification. La liste de missions à effectuer évolue donc en fonction du temps.

Le critère d'optimisation considéré dans ce cas d'étude pour prendre les meilleures décisions est le revenu d'exploitation qui inclut les coûts de maintenance ainsi que les coûts d'utilisation associés à l'exécution des missions et aux potentiels retards de livraison.

Une étude comparative entre l'algorithme dynamique et un algorithme dit « statique » est menée. Cet algorithme statique ne replanifie pas en cas de panne ou d'information de dégradation. Dans le cas où une nouvelle mission est disponible, elle est mise en attente et ne sera planifiée que lorsque toutes les missions déjà planifiées auront été exécutées. L'algorithme

de planification dynamique permet d'améliorer les revenus d'exploitation d'environ 9% .

Ces différentes études montrent l'importance d'intégrer les informations temps-réel dans le processus de prise de décision afin de mettre à jour le planning conjoint des maintenances et des missions. Replanifier permet d'augmenter le revenu d'exploitation du véhicule mais la gestion de la replanification doit se faire avec soin afin d'éviter de replanifier trop souvent, ce qui créerait une instabilité du planning. On ne pourrait pas s'y fier sur le long terme étant donné qu'il serait constamment amené à évoluer. Il est donc question de trouver un équilibre entre le nombre de replanification et la criticité des informations utilisées pour déclencher une mise à jour du planning.

La prochaine étape est d'inclure la dimension flotte dans l'algorithme de prise de décision pour tirer parti de la disponibilité de tous les véhicules de flottes ainsi que de leurs différentes configurations (type de moteurs, cycle de transport pour lesquels ils sont définis, ...).

Chapitre 7 : Planification conjointe des missions et des opérations de maintenance pour une flotte de véhicules

Ce chapitre propose deux méthodes de planification, statique et dynamique, dans le cas où des missions et des opérations de maintenance doivent être planifiées et synchronisées pour une flotte de véhicules. Il s'agit d'un problème de gestion de flotte.

Nous commençons par introduire la problématique de gestion de flotte puis présentons le problème de planification pour une flotte et les hypothèses considérées afin de présenter la méthode de résolution. La première partie est consacrée à la méthode de planification statique pour une flotte alors que la seconde présente la méthode de planification dynamique pour une flotte. Chaque partie est illustrée par un cas d'application.

7.1 Problème de la gestion de flotte

Le problème de gestion de flotte consiste à distribuer les missions entre les différents systèmes disponibles, ici les camions, de la meilleure façon possible, afin d'optimiser une fonction objective globale. Il s'agit d'une approche coopérative.

Ce problème est semblable à celui de la répartition de charges pour un système distribué. Les approches de gestion de charge sont divisées en deux ensembles : les stratégies centralisées et les stratégies décentralisées. Pour les stratégies centralisées, il est nécessaire que les informations concernant les objectifs et contraintes de chaque ressource soient disponibles pour que la tour de contrôle centralisée puisse prendre des décisions pour toutes les ressources. Ce type d'information n'est pas nécessaire dans le cas des stratégies décentralisées car il n'y a pas de coordination globale. Chaque ressource détermine les décisions à prendre basée sur les informations disponibles localement.

Dans notre cas, nous étudions une approche centralisée pour laquelle une instance prend les décisions pour toute la flotte de véhicules en se basant sur les informations de surveillance récoltées sur les différents camions.

7.2 Description du problème

Cette section décrit le problème de planification conjointe maintenance/ missions pour une flotte de véhicules dans le cas statique et dans le cas dynamique.

7.2.1 Hypothèses et contraintes

Dans le cas statique, l'ensemble des missions est connu au début de la planification et aucune information de surveillance n'est disponible. Au contraire, dans le cas dynamique, l'ensemble de missions évolue au cours du temps lorsque de nouvelles missions sont requises. Une flotte de véhicules est disponible pour exécuter les missions. Chaque véhicule se détériore et cette dégradation est différente d'un véhicule à l'autre en fonction des missions exécutées et de leurs sévérités d'usage. Le processus de dégradation est toujours modélisé à l'aide d'un processus Gamma.

Cette flotte peut être composée de véhicules ayant différentes configurations, adaptées à des usages spécifiques, avec des composants différents installés sur le véhicule, comme le moteur. C'est pourquoi, l'impact d'une même mission sur la dégradation de deux véhicules ayant différentes configurations peut changer. Une nouvelle composante à la définition d'une mission est ajoutée. Deux ensembles de paramètres pour le processus Gamma sont associés à chaque mission. Chaque ensemble permet de décrire l'évolution de la dégradation dans le cas où le camion a une configuration de type 1 ou de type 2.

7.2.2 Objectif

L'objectif est de définir un algorithme permettant de planifier conjointement les maintenances et les missions pour toute une flotte de véhicules. Le cas statique permet de développer une méthodologie définissant la base de la méthodologie adaptative dans le cas dynamique.

Le planning global pour la flotte est composé de sous-plannings, un pour chaque véhicule. Chaque sous-planning est défini par des blocs de missions, séparés par des opérations de maintenance. Ces opérations sont planifiées de façon optimisée en prenant en compte le modèle d'évolution de la dégradation pour chaque véhicule.

On attend de la méthodologie qu'elle définisse un processus de décision pour assigner les missions aux différents véhicules et planifier les opérations de maintenance en conséquence.

Dans le cas de la planification dynamique, la stratégie à développer est une stratégie prédictive-réactive qui ajuste le planning en fonction des événements tels que les pannes, les mesures de dégradations et les nouvelles missions disponibles.

Il est à noter qu'inclure la dimension flotte dans la prise de décision n'implique pas de reprendre l'étude du problème de planification conjointe à zéro. Il s'agit d'adapter les méthodes déjà développées dans les cas statique et dynamique pour un seul véhicule.

7.3 Rappel : modèle de dégradation du véhicule

On rappelle que la dégradation de chaque véhicule est caractérisée par un indicateur global

de dégradation qui augmente graduellement en fonction du temps, tant qu'aucune opération de maintenance n'est réalisée. Le processus Gamma utilisé pour modéliser l'évolution de la dégradation est un processus à paramètres variables en raison du changement d'environnements opérationnels liés aux différentes conditions de sévérité associées aux missions.

Il est à noter que la même mission attribuée à deux véhicules de différentes configurations n'aura pas le même impact sur leurs dégradations.

Toutes les opérations de maintenance sont considérées comme parfaites et rétablissent le niveau de dégradation du véhicule à zéro.

7.4 Le cas statique

L'intérêt d'étudier le cas statique est de voir les avantages offerts par la capacité de la flotte pour planifier et exécuter toutes les missions demandées. En effet, considérer une flotte apporte de la flexibilité pour assigner les missions aux véhicules ayant le niveau de santé le plus adéquat.

7.4.1 Description de la méthode de résolution

L'approche pour résoudre le problème de planification conjointe dans le cas d'une flotte comporte de nombreuses similarités à celle permettant de résoudre le même problème pour un seul véhicule. Dans le cas de flotte, on ajoute un degré de complexité supplémentaire car il ne s'agit plus seulement de planifier les missions dans le bon ordre et les maintenances au bon moment mais également d'assigner les missions aux véhicules adéquats.

7.4.1.1 Critère de décision

Le critère de décision choisi dans le cas statique est celui du coût de maintenance global pour la flotte. Il est obtenu en sommant le coût de maintenance estimé pour chaque véhicule de la flotte. Ce coût de maintenance considère les coûts préventifs associés aux maintenances planifiées dans le planning et les coûts correctifs associés aux probabilités d'avoir des pannes dans les blocs de missions définissant le planning.

7.4.1.2 Algorithme génétique

La méthode utilisée pour résoudre ce problème d'optimisation est basée sur un algorithme génétique. Il s'agit d'une adaptation de l'algorithme génétique développé dans le cas statique pour un seul véhicule en considérant qu'un individu est un planning global pour la flotte. Ce planning global est composé de sous-plannings qui définissent chacun le planning pour un véhicule de la flotte.

7.4.2 Analyse des performances

Pour analyser les performances de l'algorithme pour la flotte, nous allons comparer les résultats obtenus si nous planifions et exécutons un ensemble de missions en utilisant l'algorithme statique pour un seul véhicule. Cette méthode, appelée méthode 1VS1, consiste dans un premier temps à distribuer les différentes missions entre les différents camions. Ensuite,

l'algorithme de planification pour un véhicule est appliqué à chaque camion indépendamment pour planifier les missions qui lui sont attribuées.

Dans le cas d'une flotte homogène, l'algorithme pour la flotte permet de réduire le coût de maintenance global de la flotte de 2 à 6,5% en fonction de la valeur de la condition de remplissage des blocs \mathbb{P}_{max} . Le maximum est atteint pour $\mathbb{P}_{max} = 0.1$. Dans la plupart des cas d'étude, la méthode utilisant la dimension flotte permet de réduire le coût de maintenance total pour la flotte. Lorsque ce n'est pas le cas, cela est dû à la valeur de \mathbb{P}_{max} . En effet, plus elle augmente, plus la contrainte de remplissage des blocs de missions est relaxée. L'algorithme utilisant la dimension flotte prend alors plus de risque pour définir un planning que la méthode 1VS1. Dans le cas d'une flotte hétérogène avec deux types de configuration pour les véhicules, les économies réalisées sur le coût de maintenance global en utilisant la dimension flotte augmentent. Cette méthode permet de réduire le coût de maintenance global d'environ 25%.

7.5 Le cas dynamique

La méthode mise en place permet d'adapter le processus de décision afin d'intégrer les informations temps réel venant des véhicules de la flotte pour mettre à jour le planning complet de la flotte, c'est-à-dire le planning de chaque véhicule. Nous considérons trois événements temps réel : les occurrences de pannes, les mesures de dégradation et les nouvelles missions demandées.

L'objectif est de mélanger les aspects flotte et dynamique pour prendre les meilleures décisions possibles pour assigner les missions aux bons véhicules et planifier les opérations de maintenance en fonction de l'état de santé des véhicules.

7.5.1 Description de la méthode de résolution

La méthode de résolution est basée sur la méthode dynamique développée pour un seul véhicule en intégrant des aspects relatifs à la flotte, définis dans le cas statique pour la flotte.

7.5.1.1 Définition des caractéristiques des missions

Les missions sont caractérisées par des conditions opérationnelles spécifiques qui représentent leur sévérité d'usage. Cela permet de définir leur impact sur la dégradation des véhicules. Comme les véhicules peuvent avoir différentes configurations, la sévérité d'usage d'une mission diffère d'une configuration à une autre. Ces caractéristiques sont décrites à l'aide d'une paire de paramètres permettant de définir le processus suivi par la dégradation du véhicule. Il y a une paire de paramètres pour chaque configuration véhicule.

Une mission est également définie par sa durée, son gain, le coût de retard unitaire et sa date limite d'exécution, c'est-à-dire la date avant laquelle elle doit être terminée.

7.5.1.2 Critère de décision

Le processus de décision pour planifier les maintenances et les missions pour la flotte de véhicules cherche à maximiser les revenus d'exploitation pour toute la flotte. Il s'agit donc d'un objectif commun. Il prend en compte les gains générés par l'exécution des missions, les

coûts de retard si les missions ne sont pas terminées avant leurs dates limites, les coûts de maintenance et un coût associé au chargement/ déchargement de marchandises dans le cas où une mission en cours est attribuée à un autre véhicule durant une replanification. Ce dernier coût de pénalité permet d'éviter d'avoir un planning trop instable et de modifier trop souvent le schéma d'attribution des missions aux véhicules.

7.5.1.3 Description de la méthode de replanification pour la flotte

La méthode de replanification est basée sur une combinaison des algorithmes génétiques développés pour le cas dynamique avec un véhicule et le cas statique avec une flotte.

Au départ, un planning initial pour la flotte est généré à l'aide de l'algorithme génétique. L'exécution de ce planning commence et des replanifications peuvent se produire lors des occurrences de trois événements : les pannes, les mesures de dégradation disponibles et les nouvelles missions. Le principe des replanifications est similaire à celui décrit dans le cas dynamique pour un véhicule. Cependant, la replanification affecte toute la flotte, peu importe l'activité en cours pour chaque véhicule à ce moment-ci. Nous supposons que les véhicules peuvent alors stopper leur mission en cours et se voir assigner une nouvelle mission s'il s'agit de la meilleure décision pour améliorer les revenus d'exploitation de la flotte. L'intérêt de prendre des décisions au niveau de la flotte de véhicules, même si l'information disponible ne concerne qu'un seul véhicule est que l'on peut éviter de potentiels retards dans les livraisons, ce qui améliore les revenus d'exploitation. De même, si une information de dégradation permet de conclure que le véhicule ne peut pas terminer sa mission en cours, replanifier toutes les missions pour toute la flotte peut permettre d'identifier une autre mission, déjà assignée à un autre véhicule, qu'il pourrait faire avant d'être maintenu. Cela permet d'utiliser le véhicule au maximum de sa durée de vie.

L'algorithme développé permet de considérer tous les événements de façon simultanée, qu'ils se produisent sur un ou plusieurs véhicules. Différent cas de figures existent en fonction de l'activité en cours des véhicules. Dans le cas d'une replanification, tous les véhicules déployés sur une mission arrêtent leur activité et attendent les nouveaux ordres. Ils sont considérés comme étant disponibles dès qu'une replanification a lieu. En revanche, si des véhicules sont en maintenance, ils sont considérés comme non disponibles. Ils ne seront à nouveau disponibles que lorsque leur opération de maintenance sera achevée. L'algorithme prend en compte cette période de non-disponibilité pour construire le nouveau planning optimisé.

7.5.2 Exemple d'application pour analyser les performances de la méthode de planification dynamique pour la flotte

De même que dans le cas statique, nous comparons la méthode dynamique de planification intégrant la dimension flotte avec la méthode dynamique de planification pour un véhicule, appliquée à chaque véhicule indépendamment. Cela signifie que la première étape est d'assigner les missions aux véhicules. Ensuite, nous considérons que les véhicules sont complètement isolés les uns des autres et un événement temps réel engendre donc uniquement la mise à jour du planning du véhicule concerné. Les véhicules ne peuvent donc pas échanger des missions

entre eux. Il s'agit donc de comparer une approche centralisée, intégrant la dimension flotte dans le processus de décision, à une approche décentralisée, où la prise de décision se fait pour chaque véhicule indépendamment.

Les différents cas d'études montrent qu'utiliser la méthode dynamique intégrant la dimension flotte permet d'augmenter les revenus d'exploitation pour la flotte. Dans le cas où le nombre de replanification pour chaque méthode est similaire (Section 7.5.2.2 – étude 1), il y a davantage de replanifications dues à des pannes avec la méthode décentralisée. Les informations de dégradation sont également plus utiles pour la méthode intégrant la dimension flotte. Dans l'étude 2, le nombre de replanification est plus grand pour la méthode centralisée. Cela vient du fait qu'il y a beaucoup plus de maintenances préventives dans les plannings obtenus avec la méthode décentralisée. La méthode centralisée prend donc plus de risque d'avoir des pannes. En revanche, dans la majorité des simulations, les coûts de retards pour la méthode centralisée sont plus faibles que ceux pour la méthode décentralisée. La méthode centralisée permet globalement d'améliorer les revenus d'exploitation de la flotte d'un peu plus de 6%.

Ces résultats sont prometteurs mais il faut rester vigilant lorsqu'il s'agit de replanifier car des replanifications trop fréquentes pourraient se révéler moins profitables d'un point de vue logistique. En effet, des échanges de missions entre les différents véhicules peuvent engendrer un coût non négligeable pour le chargement et le déchargement des marchandises d'un véhicule à un autre.

Néanmoins, les résultats montrent que le développement d'une solution de gestion de flotte basée sur une optimisation conjointe du planning de maintenance et du planning de missions offre de nombreuses opportunités pour améliorer la rentabilité des clients. Ce serait un excellent outil d'aide à la décision pour améliorer les revenus d'exploitation tout en utilisant au maximum la capacité de la flotte.

Chapitre 8 : Conclusions et perspectives

Ce travail de recherche présente les différentes contributions permettant d'apporter une réponse au problème de planification conjointe maintenance/ missions pour un seul véhicule ou pour une flotte de véhicules. Nous avons mis l'accent sur trois principaux axes de réflexion : comment planifier de façon conjointe les maintenances et les missions pour un véhicule dont l'état de santé se dégrade, comment définir une méthode intégrant les informations temps réel pour mettre à jour le planning et comment intégrer la dimension flotte dans le processus de prise de décision si nous devons gérer l'activité de plus d'un véhicule.

L'objectif de cette thèse est de développer une méthodologie reliant le planning de maintenance et les aspects de gestion de flotte liés à la planification des livraisons et d'évaluer la valeur ajoutée qu'une telle méthode peut apporter au client pour l'aider à prendre des décisions. Un service de gestion de flotte mis en place comme outil d'aide à la décision pour le client est vu comme un outil utile pour améliorer la rentabilité et la satisfaction du client.

De nombreuses perspectives, court-terme comme long-terme, peuvent être évoquées afin de continuer ce travail de recherche.

Parmi les perspectives court-terme, la première serait de continuer le développement de la méthode dynamique de planification pour la flotte. La seconde serait de retirer les hypothèses simplificatrices notamment celle considérant que les paramètres du processus de dégradation associés à chaque mission sont parfaitement connus. La dernière perspective serait d'intégrer la dimension géographique dans le processus de décision qui constitue également un point central permettant de choisir entre déployer un véhicule sur une mission et sur quelle mission ou l'envoyer en atelier de maintenance.

Les perspectives long-terme sont principalement centrées sur des phases de tests des différentes méthodes sur des cas concrets de gestion de flotte. En effet, jusqu'ici, l'analyse des méthodes développées s'est faite uniquement en simulation mais pas sur des cas réels. C'est un point fondamental pour évaluer la valeur ajoutée offerte par l'implémentation de cette méthodologie. De plus, les clients sont dans l'attente de services leur offrant des solutions et des recommandations sur la maintenance mais également sur l'usage opérationnel pour garder leurs véhicules aussi opérationnels que possible. L'objectif du Groupe Volvo est donc de développer davantage les méthodes de prises de décisions post-pronostic afin d'aider à renforcer le passage entre la prédiction de la dégradation et le calcul de la durée de vie résiduelle d'un système à la prescription d'actions optimales pour le système, en termes de mode d'utilisation ou de maintenance. L'objectif est de proposer un processus décisionnel complet basé sur les données connectées concernant l'état de santé du véhicule, ses conditions d'utilisation, sa localisation spatiale et ses contraintes d'exploitation. Il ne s'agit plus seulement de planifier la maintenance en fonction de la dégradation du véhicule, mais aussi d'intégrer la possibilité de modifier les réglages du véhicule afin qu'il puisse poursuivre sa mission en cours, ou attendre le meilleur moment et le meilleur endroit pour effectuer une réparation. La décision post-pronostic et les prescriptions qui en découlent peuvent donc concerner à la fois l'entretien et/ou un changement d'utilisation, voire une adaptation de la stratégie de surveillance lorsque l'état de santé est trop dégradé.

General introduction

Contents

1.1 Maintenance management for industrial vehicles	2
1.1.1 Main contributors to the maintenance activity	4
1.1.2 How does Volvo Group build a maintenance contract?	5
1.1.3 How to define the maintenance operations schedule in the Volvo Group? .	7
1.2 Towards an improvement in the service offers	8
1.2.1 Customers needs	9
1.2.2 Considering the technological and societal evolutions	10
1.2.3 Contributions to improve the offered services and their limitations	12
1.3 Thesis contribution	17
1.3.1 Industrial objectives	17
1.3.2 Methodological challenges and thesis contribution	19
1.4 Thesis outline	21

This first chapter introduces the issues related to the management of the maintenance operations and the delivery activity for fleet of industrial vehicles. The objective is to present the actual position of the Volvo group regarding these questions and how it is going to evolve over time.

The first part deals with the vehicles maintenance management. We start by presenting the different stakeholders involved in the maintenance activity. Then, we explain how a maintenance contract and a maintenance operations schedule are defined for a truck.

The second part details the services offered by the Volvo group and how they are going to evolve. The first stage to complete is to analyse the customers needs when it comes to service. Then, we also need to study the changes that appeared in the past few years regarding the business context to adapt the potential future services to the market. The last stage briefly describes the initiatives launched to help answering the customers needs and their limits.

This statement enables to make a connection with the last two sections that describe the contributions brought by this PhD research work and the structure of the thesis.

1.1 Maintenance management for industrial vehicles

In 2018, the road freight transport represented 50.6% of the freight transport in the whole Europe (Figure 1.1). Indeed, this transport way is relatively cheap and very flexible and can even be used for door-to-door deliveries. The other transport modes are not as satisfying as road transport. Rail transport is much more inflexible and limited by the rail networks and terminals. But it still quite cheap and can transport bulk quantities. Sea transport is very cost effective for heavy goods but also very slow and limited by ports, shipping channels and canals. Air transport is fast and secure but very expensive. It explains why it represents only 0.1%tkm because it is generally used for valuable or urgent deliveries.

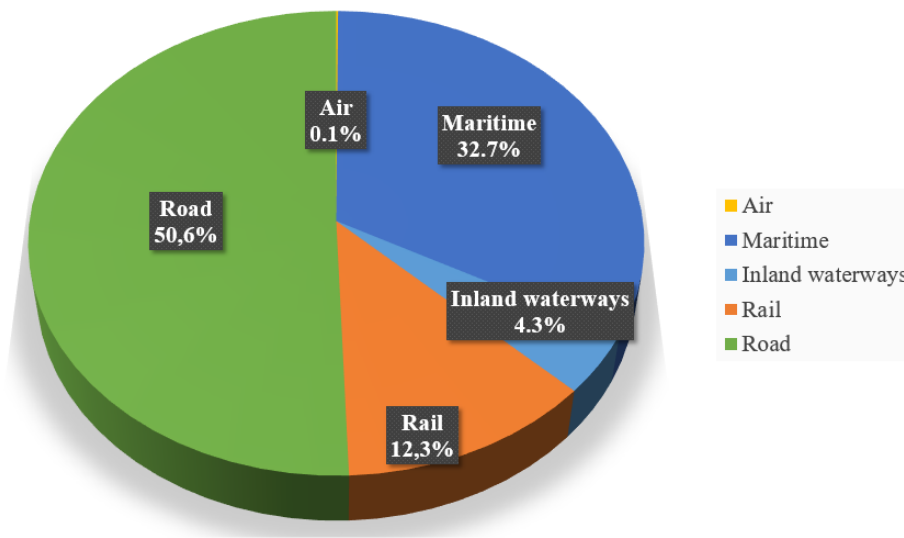


Figure 1.1: Distribution of the freight transport modes in Europe in 2018 (%tkm) [160]

The portion representing the road freight transport has been quite stable for the past ten years and shows that road transport is still favoured with respect to the other transportation modes. It is the less expensive and the most flexible way to transport a huge capacity of goods.

However, the development of new technologies has started to change the road transport business model and new emerging trends are about to transform the whole transport solution design and the way they are used. These trends are digitalization with the development of connectivity, automation and electromobility. Electromobility has become a top priority owing to the drastic measures taken to reduce CO_2 emissions. The European Union agrees on a 30% reduction in CO_2 emissions from trucks by 2030 with an intermediate target of 15% by 2025 [110].

The trucks are considered as a working tool to earn profit. Over the years, the haulers have been trying to reduce as much as possible the operating costs by using their trucks at the maximum of their capacity and availability. They are working with just-in-time flow. Managing the global operating incomes for a fleet becomes essential to improve the customers

profits. To do so, we have to identify the main sources influencing the operating incomes.

The French National Road Committee delivers some statistics every year to observe the evolution of the working methods of road haulage for goods markets. Figure 1.2 describes the distribution of the operating expenses for a long haul vehicle over a year in 2018 [43]. Long haul has to be understood as domestic or international carriage whose operating constraints make it impossible or uncertain for the driver to go back home daily.

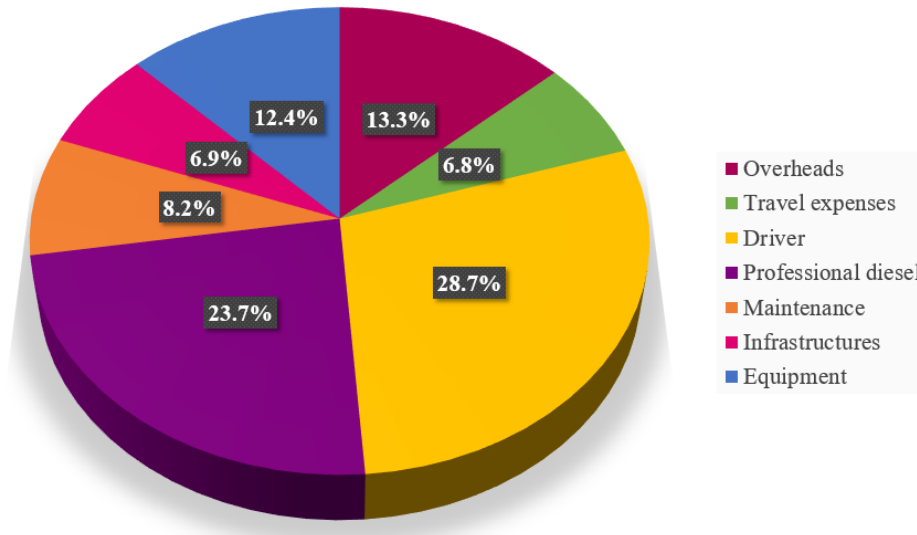


Figure 1.2: Annual operating expenses for a long haul vehicle (2018)

The different expense centres are defined as follows:

- **Driver:** the cost of drivers gathering conventional scale evolutions, wage charges and consequences of social regulation modifications.
- **Professional diesel:** the fuel cost index before VAT, taking into account various refuelling methods (pump and storage tank) and the partial refunding of the French Energy Tax.
- **Overheads:** structural costs and other indirect charges composed of moderate administrative staff costs.
- **Equipment:** the costs of possession, renewal and financing (according to different methods: borrowing, leasing, renting) and insurance costs (vehicle and transported goods).
- **Maintenance:** the maintenance and repair costs as well as tyre costs.
- **Infrastructures:** the costs of motorway tolls and the axle tax.
- **Travel expenses:** the cost of a long distance working day including two meals, one accommodation and one snack (based on the Labour Agreement and the travel expenses protocol).

The highest expenses are dedicated to pay the drivers and the fuel. The truck manufacturer needs to focus on the expenses directly related to the vehicles such as the fuel consumption and maintenance. The maintenance activity represents 8.2% of the operating expenses for a vehicle. It cannot be neglected so it is essential to focus on the maintenance management. Two main reasons can be mentioned. The first one is to ensure that the vehicle is correctly operating. The second one is that, if the maintenance operations are performed at the appropriate moments, it is possible to increase the vehicle availability and then increase its operating duration to generate more profit.

In addition, the operating expenses, described in [Figure 1.2](#), do not consider the potential delay costs associated with a delay on the delivery. The costs associated with the maintenance activity are then underestimated. Working on the maintenance activity for industrial vehicles seems like a promising topic to reduce the operating expenses.

1.1.1 Main contributors to the maintenance activity

Before detailing how the maintenance activity is handled in the Volvo Group, it is necessary to define the different groups of people involved in this field [97].

- **Transport purchaser:** He is expecting a high quality transport service that allows him to be delivered at the right time and at an attractive price.
- **Hauler:** He is the one buying the transport solution. His objective is to satisfy the transport purchaser needs by ensuring an effective and competitive transport service offer. He then needs to secure the quality of his offer without generating too high operating costs. He employs the driver and keeps him updated on the missions to complete and the maintenance planning.
- **Driver:** He has to make sure that the vehicle is always operating at its defined performance. He has to respect the maintenance planning and alert the hauler if there is any sign of a vehicle component failure. He is also in contact with the workshop if he needs information about the vehicle health state.
- **Workshop:** It is where the maintenance operations are performed. The mechanics working there have to perform the maintenance actions according to the defined maintenance planning. The workshop also considers the unplanned immobilizations due to a breakdown or an accident. Its objective is to perform the operations fast and well to reduce the vehicle downtime. The employees are in contact with the driver, the hauler and also the suppliers to manage the spare parts orders.
- **Supplier:** His role is to deliver the spare parts for the vehicle to the workshop to ensure the repair. Besides the current restocking, he may have to perform urgent deliveries when a vehicle is subject to an unplanned stop to avoid a too long vehicle unavailability period.

- **Dealer:** His role is to satisfy his customers by proposing vehicles and associated services to fit their needs. The maintenance contract is part of the services suggested to the customers. It is defined when the customer purchases the vehicle. The dealer also requires the hauler to define the contract to suit the vehicle usage.
- **Governmental authorities:** They impose legal rules to control the road freight transport. These rules can affect the maintenance activity, the safety, and the surrounding infrastructures. For instance, some regulations regarding the NO_x and CO_2 emissions are defined to decrease air pollution. The rules enable to outline the road freight transport framework.

1.1.2 How does Volvo Group build a maintenance contract?

This section describes how the Volvo Group defines a maintenance contract for a vehicle sold to a customer. It is only based on the Volvo Group experience and expertise.

The dealer offers to the hauler the possibility to subscribe to a maintenance contract. This contract aims at ensuring the vehicle availability by managing its maintenance activity. For the vehicle user, this contract is like a warranty that the vehicle keeps a constant performance level over time. The definition of the contract depends on both the vehicle configuration and its expected usage. The vehicle configuration is based on the types of systems and components set up on the vehicle.

When the vehicle is purchased, the user specifies the future usage conditions. To do so, the dealer and the user evaluate some variables related to the transport cycle (distribution, long haulage, ...), the operating environment (topography, state of the roads, type of roads like highways or in agglomeration, ...), the vehicle usage (annual mileage, speed changes, ...). Combining the different elements describing the vehicle usage enables to select the adapted maintenance intervals for the maintenance operations of each component.

The approach begins with defining the Powertrain Operating Conditions (POC) of the vehicle. The POC describes the way of use for the vehicle (Figure 1.3). Based on its transport cycle, its gross combined weight (GCW) i.e. the weight of the tractor and the trailer, and the topography, the POC is defined. Then, the corresponding maintenance interval for each component can be selected.

Transport Cycle	Long distance							Distribution regional			Distribution city	Construction			
GCW	$\leq w_1$		$w_1 - w_2$		$w_2 - w_3$		$> w_3$	$\leq w_4$	$w_4 - w_5$	$w_5 - w_6$	$\leq w_7$	$\leq w_8$	$w_8 - w_9$	$w_9 - w_{10}$	$> w_{10}$
Topography	PF	H	PF	H	PF	H	-	-	-	-	-	H	H	H	VH
POC	L	M	H	S	S	VS	VS	M	H	S	S	H	S	VS	VS+

Figure 1.3: Powertrain Operating Conditions (POC) defined by the Volvo Group for a specific truck model

where:

- Topography abbreviations:
 - PF = Predominantly Flat
 - H = Hilly
 - VH = Very Hilly
- POC abbreviations:
 - L = Light
 - M = Medium
 - H = High
 - S = Severe
 - VS = Very Severe
 - VS+ = Extreme

A maintenance interval for a component is a static regular interval respecting the component technical specifications (Figure 1.4). It defines the optimal or recommended replacement age for the component. Different elements are used to define an interval. It can either be a fixed time interval according to the calendar time (number of months) or a duration of use (number of engine hours) or a usage distance (mileage). For each POC value, an interval is specified for each component and each operation. The selected interval corresponds to the minimal value between the three features according to the vehicle use. Naturally, if the vehicle is immobilized owing to a failure occurrence, a maintenance operation is also carried out.

For most components, the intervals defined by the company are obtained with the help of durability tests and a study of the maintenance cost. The durability tests are endurance tests completed in test cells. The component is tested on its own, independently from the system, and with constraints depending on the specific usage. It enables to acquire the a priori survival function for the component according to this specific usage [31]. A cost function can then be built based on the replacement cost and the a priori failure function. The optimal replacement age corresponds to the value minimizing the cost function. Sometimes, the durability tests are too complex to set up. So, experience and technical knowledge on the vehicle enable to define the recommended replacement ages.

All the definitions for the POC and the components maintenance intervals for the trucks are listed in a service bulletin called « Preventive Maintenance Intervals » [136].

Based on the maintenance intervals for the components, a schedule is built to inform the vehicle user when the maintenance operations have to take place.

POC		L	M	H	S	VS	VS+
System/ Components		Mileage / months / hours					
21413-2	Operation 1	First interval only: 200 000 / 24 / -		First interval only: 200 000 / 24 / - subsequent intervals: 400 000 / 24 / -		200 000 / 12 / 4 000	
26068-3	Operation 2: Config 1	500 000 / 36 / 8 000			400 000 / 36 / 6 600	300 000 / 36 / 5 000	200 000 / 36 / 3 300
	Operation 2: Config 2	300 000 / 36 / 5 000				200 000 / 24 / 4 000	
25477-3	Operation 3	500 000 / - / -	375 000 / - / -	300 000 / - / 4 500	225 000 / - / 4 500	100 000 / - / 4500	- / - / 4 500
25627-3	Operation 4	- / 24 / -				- / 12 / 4 000	

Figure 1.4: Maintenance intervals for components related to a specific system installed on a specific truck model

1.1.3 How to define the maintenance operations schedule in the Volvo Group?

Once the maintenance schedule has been defined, based on the replacement intervals for all the components supported by the contract, the customer has to choose which schedule service he would like between two modes: the « static » and the « adaptive » ones [97].

With the « static » mode, the features, defined when the vehicle is purchased, on its future usage and configuration are used to schedule the maintenance operations. The scheduling algorithm mainly uses the annual mileage, the annual number of engine hours and information on the operating conditions. There is no update considered on the given information at the purchase. The maintenance schedule is then never updated. That is why, this schedule mode

is called « static ».

The « adaptive » mode automatically updates the maintenance schedule when a change is detected in the vehicle usage . This change can either be on the conditions of use or on the mileage or engine hours. The scheduling algorithm then recomputes the annual average mileage and the annual average number of engine hours on a 6-month rolling horizon. To update the operating conditions, the fuel consumption is used. A requirement to subscribe to this mode is for the vehicle to be connected via the telematic gateway to retrieve the monitoring data.

The only difference between the two modes are only related to the potential updates of the maintenance intervals. The way to schedule the operations is similar. The objective is naturally to avoid unplanned stops but also to avoid having too many maintenance operations at different dates. Some operations will then be grouped. In a first step, the annual or periodic workshop visits are scheduled. They fix the basics for scheduling the other operations. Then, the most frequent operations are added to the schedule and are fixed according to the previous defined visits to group at best the operations. If there are maintenance operations only defined by a maintenance interval in months, they are scheduled after the annual, periodic and most frequent visits. Finally, the remaining operations are scheduled based on the visits dates for the previous ones.

The maintenance operations grouping is obtained by considering a fixed time window. The schedule is browsed and if there are several operations in the time window, they are grouped. The date of the workshop visit is updated to correspond to the average date between the first and the last operations dates in the window. The purpose of this visits dates update strategy is to prevent the user from coming to the workshop too often and therefore, reduce the availability of his vehicle.

1.2 Towards an improvement in the service offers

Volvo Trucks and Renault Truck, two brands from the Volvo Group, propose two major types of services when it comes to manage the trucks they sell.

- **Maintenance contract:**

The first one is the maintenance contract to offer a suitable maintenance management solution to the customer to ensure the vehicle availability and avoid unplanned stops. Using preventive maintenance models for the components and scheduling the workshop visits really enable to reduce the immobilization time and the maintenance time.

- **Fleet management:**

The second type of service is related to fleet management. The objective is to offer a turn-key service to the customer to evaluate the state of his fleet of vehicles. For the brand Volvo Trucks, this service is called « Dynafleet » [164]. It is composed of three

to four packages to help the customer make the best decisions to manage his fleet. The packages are defined as follows:

- **Fuel consumption and environment:** This package enables to follow the vehicle performance and the driver efficiency. The fuel consumption, the covered distance as well as the CO_2 emissions for the trucks are available. It analyses the performances of each driver regarding its energetic footprint and gives drivers advice to optimize their way of driving. An alert system informs the customer about any significant variation of the fuel level. Reports are also available to offer potential improvements to reduce the costs and favour sustainable savings.
- **Driver times:** This service automatically downloads all the tachograph and digital driver card data to build prognostics and monitor the driving times of the drivers. It enables to identify which driver is the most likely to take a new mission, who must take a break or even stop working for the day.
- **Fleet location:** This package allows the fleet manager to follow all the vehicles from his park. He has access to the GPS position of all the vehicles, information related to the vehicles types, loads, speed.
- **Messaging:** With this service, the office can communicate easily and directly with the driver to avoid any misunderstanding and can help the driver make the best decisions.

The fleet service « Optifleet » [163] for the brand Renault Trucks proposes similar offerings. In both cases, the objective is to help the customer generate profits by indicating them at which level decisions can be made to improve their profitability.

With an evolution of both the customers needs and the changes brought by a technology shift in the industry and the society, the offered services have to be improved to meet the customers expectations and integrate the new possibilities offered by the new technology.

1.2.1 Customers needs

To design the right service offers, it is necessary to study and understand the customers needs. For an hauler, a truck is a working tool. That is why, he is interested in a complete transport solution including both the vehicle and the services adapted to the real usage of the vehicle. Naturally, this transport solution must suit to the hauler strategy. Actually, the maintenance management is among the most expected services. Indeed, to maximize the profit generated by the vehicle activity, one of the objectives is to minimize the operating costs by optimizing the maintenance schedule.

The first expectation expressed by the users is to have access to flexible services that fit to their strategy. It is all the more valid for maintenance management services. Most of the haulers own several industrial vehicles. This vehicle set is called a fleet or a park. In this case, handling the uncertainty is essential. Due to the economic constraints, they work on a just-in-time mode. According to the fleet size, the strategy can be different.

In fact, for a small fleet, the main objective is to complete the mission assigned to the vehicle until the goods delivery. The vehicle maintenance actions then have to be scheduled in advance to manage at best the fleet use as well as the allocation of the missions to the vehicles. We must ensure that no breakdown or vehicle immobilization will occur during the mission completion. The fleet manager will then aim at increasing the vehicle uptime and decreasing the downtime. The uptime refers to the operational availability i.e. the time during which the vehicle is completing missions while the downtime indicates the time during which the vehicle is unavailable due to a breakdown or a maintenance operation.

If the hauler owns many vehicles, he can handle a higher flexibility when it comes to manage his fleet. It is then possible to increase the duration between two maintenance operations. In addition, if an immobilization for a vehicle occurs during a mission, this vehicle can be replaced by another operational one to finish the current mission. Companies having larger fleet are more focused on the global maintenance cost spent to maintain their vehicles. They aim at minimizing this cost to increase their profit.

Whether the customer owns a small or a big fleet of vehicles, they still have close objectives. On one hand, increasing the vehicles uptime implies that we want to minimize the number of scheduled stops on a defined time horizon and naturally avoid the number of unplanned stops. On the other hand, minimizing the vehicle maintenance cost suggests that the maintenance operations are performed at the right time but also that we avoid vehicle immobilizations that lead to expensive additional costs. In both cases, the objective is to avoid the vehicles breakdowns or any unnecessary operation that can lead to an immobilization. This is the second expectation for the customers.

The third expectation is to be able to ensure the deliveries while respecting the deadlines and avoid any potential penalty costs. Indeed, the first objective for an hauler is to ensure an effective delivery of the goods at the right time to its customers. Managing the fleet of vehicles then implies to schedule the maintenance operations at the right time but also to ensure that vehicles are available at the right time with a good enough health state to respect the operational constraints i.e. to complete the missions before the deadlines. Considering both aspects will enable to minimize the delay costs associated to the missions completion. This expectation raises the question of the definition of a mission. This definition will be clarified in [Section 1.3](#).

1.2.2 Considering the technological and societal evolutions

The new service offerings naturally have to be designed according to the customers needs. But there is another dimension that must be considered when thinking about the future services. We live in a world that is constantly changing and evolving. New technologies have seen the light in the past few years and we need to integrate them to stay competitive and in phase with the future. Three major trends are shaping our future society and will greatly impact mobility and have potential to transform transport solutions. These three axes are digitalization, automation and electromobility [143].

- **Digitalization:**

Digitalization is based on different new technologies such as connectivity, the Internet of things, artificial intelligence and data analytics [72]. They are all leading to new services and products and opening up new opportunities. Connectivity is probably the first new technology that appeared in the past ten years, around the year 2012, but its deployment and use has been growing since then. Just about every device will be connected and able to collect data and this change also includes vehicles. Nowadays, a lot of data is collected regarding the components operations, the operating conditions, the environment, the vehicle health condition, the fuel consumption, the driving conditions, the occurring failures and so on. Connected vehicle technology provides seamless connectivity through cellular systems, short-range vehicles-to-vehicles and vehicle-to-infrastructure solutions. It enables the development of on-board platforms, back-office logging and monitoring systems to collect data but also satellite positioning systems and map applications. The data collected through connectivity can help design algorithms for usage modelling, failure predictions, diagnostics and service planning tool to increase and optimize the vehicles uptime [112].

- **Automation:**

Automation has the potential to significantly improve both productivity and safety and will offer new possibilities when it comes to services and transport solutions [72]. For driver-less transport solutions, services to help make decisions for fleet and maintenance management will be necessary to know whether or not a vehicle can be deployed on a selected mission or needs to be send to the closest workshop for repair. Trucks will also increasingly need to communicate and cooperate with other vehicles or systems and with the infrastructures. Developing automation changes what machines and vehicles look like. The mining and construction sectors are the leaders when it comes to develop automated solutions. The three main advantages are focused on safety by eliminating driver error and reducing the number of people being around vehicles or in dangerous environments, on productivity because work can be carried out more efficiently over longer hours, and on work life as operators will be assigned to more supervisory duties [77].

- **Electromobility:**

Nowadays, reducing our CO_2 emissions has become a top priority to protect the planet and live in cleaner cities. Electromobility is a critical technology to master to help build more energy-efficient societies. The first full electric vehicles developed by the Volvo Group are mainly driving in the cities. It is either city buses or urban distribution vehicles such as refuse trucks for waste management. However, technological advances, especially in energy infrastructure, storage and battery charging, will gradually reduce costs and improve performance, making electromobility concepts more feasible, even for heavy vehicles [72]. That is why, the first projects are launched in 2019 in the group to manufacture full electric long haul trucks. The electric trucks could have fewer part replacements and lower maintenance costs insofar as an electric engine has about 20 moving parts while a conventional internal combustion drivetrain has around

200 [148]. But it does not change the fact that fleet and maintenance management services will still be needed to handle the maintenance operations for the remaining components. The major concern for customer about electromobility is certainly the vehicle autonomy. As we are in a transition stage towards electromobility, some time will be necessary to develop a charging infrastructure as spread as for diesel and petrol stations. But it offers the possibility to develop new services based on the location of charging stations, calculating the remaining distance the vehicle can travel to handle the customers worries.

With the development of electric and automated trucks, the complete business model is changing and is more likely to be driven by a growing need for new services. Moreover, these new technologies require to have connected vehicles at any time, not only to handle the vehicle driving but also to analyse its behaviour and anticipate the needs to be sure that the vehicles can complete their missions without disturbances. This business transition has to be taken into account to design the new future services.

1.2.3 Contributions to improve the offered services and their limitations

The development of connectivity has allowed the Volvo Group to upgrade its service panel to optimize the maintenance planning definition by using data, collected thanks to the monitoring systems set up on the vehicles.

The first contribution to improve the maintenance planning took place with the European project Maintenance On Demand (MoDe) between 2009 and 2012 [115]. This project is supported by the European Union through the Seventh Framework Programme for Research and Technological Development on sustainable transport solutions. Different industrial and academic partners were involved in the project and aim at improving the vehicle availability by providing to customers appropriate services based on their vehicle configuration and missions. This service development was ensured by the construction of a maintenance platform integrating the new technologies for monitoring and for data analysis.

This project has developed three scenarios impacting the vehicle availability:

- Maintenance on Demand is focused on maintenance scheduling. The objective is to integrate information regarding the usage conditions, the components health state and the user constraints to update and optimize the maintenance schedule. It aims at developing dynamic maintenance planning.
- Maintenance on the fly aims at improving the repair efficiency when a component health state leads to a required maintenance operation during a mission. It suggests to a user an itinerary to the closer workshop based on the vehicle location and the estimation of the remaining distance the vehicle can travel before its immobilization.
- Repair on the fly considers the case when the remaining distance does not enable the vehicle to go to the closest workshop. The developed service has then to give indications

to the driver about the safest place to park the vehicle. This place is optimized to minimize the time required for the nearest mobile workshop to act.

The Volvo Group has been working on the development of a dynamic scheduling algorithm for the maintenance operations [31]. The idea is to design a tool capable of using the dynamic information to update the maintenance schedule. These updates are based on the monitoring data on the components and the estimation of the vehicle usage conditions. To reduce the impact of the maintenance actions on the vehicle availability, a grouping strategy is integrated into the algorithm. This strategy dynamically groups the maintenance operations on a given time horizon to optimize the maintenance cost. The dynamic scheduling method is composed of four stages:

- The first stage aims at estimating the failure probability law for each component of the vehicle independently. To do so, the deterioration level is logged and a deterioration model is built based on that information to predict the future deterioration of the component. By using this model and a deterioration threshold, the failure probability law is estimated.
- The second stage uses the failure probability law to estimate the optimal maintenance date for each component by applying an age-based preventive maintenance policy. This date is a compromise between the preventive and corrective maintenance costs.
- The third stage focuses on grouping several maintenance operations at the same date to improve the vehicle availability. Grouping operations enables to reduce the number of planned stops. Hence, the logistic costs related to the vehicle entry into the workshop are reduced. However, the total maintenance cost is penalized. Indeed, some maintenance operations are planned before their optimal dates so it leads to a reduction of the useful lifetime of the components. A heuristic method has then been developed to obtain the optimal grouping structure that respects the constraints and considers the penalty cost due to the shift of the optimal maintenance dates for the different components. This structure minimizes the total maintenance cost on a given time horizon. It is composed of different maintenance operations groups whose individual and optimal replacement dates are successive.
- The last stage manages the maintenance schedule updates by fixing the operations groups and their replacement dates.

The dynamic scheduling algorithm introduces the notion of predictive maintenance. This type of maintenance enables to model the components deterioration using stochastic models and estimate the current health state of the critical components in real time. This approach presents two major advantages. The first one is that the components remaining useful lifetime can be estimated without explicitly and physically modelling the wear process. The second one is that grouping the maintenance operations according to the measured wear allows us to minimize the maintenance cost. The service offer proposed in this project enables to improve the vehicle operating availability from 15 to 30% [99].

However, this project also highlights some limits when it comes to maintain a vehicle. In fact, the interactions between the components and the system structure are not considered in the proposed maintenance policy. We cannot guarantee the best performance level for the different functions of the vehicle in that case. In addition, the maintenance operations grouping is based on the opportunities triggered by either a repair or an inspection of a component. Hence, we cannot ensure to the customers a period during which the vehicle can operate without failure insofar as we cannot predict with accuracy the moments when the system has to be maintained. Finally, in the project, the maintenance activity is not integrated to the system life cycle and no feedback regarding the vehicle manufacturing is considered. Knowledge on the failure modes and on the components deterioration are only used to optimize the maintenance decision.

In this perspective, a second research project has been led to consider the limits observed in the project MoDe [97]. The objectives include both the maintenance scheduling and the vehicle design. The approach developed in the project MoDe has to evolve to become a system approach. A system is assumed to be a set of components interacting with each other and enabling to ensure a particular function. A system approach means that the grouping strategy must be designed around the system and not the components and the maintenance decisions have to be made based on the system structure. The maintenance policy also needs to dynamically adapt to the real vehicle usage conditions. Monitoring systems have then to be installed to either collect informations about the component health state or usage conditions. Naturally, monitoring systems cannot be installed on every components for technical or economic reasons. So, the available information level is different from one component to another. Anyway, that information has to be integrated in the maintenance decision process.

The road transportation constraints have to be taken into account to design the right maintenance offer. Indeed, preventive maintenance operations are only performed when the vehicle goes to a workshop. During the missions, it is almost impossible for the vehicle to stop or it generates too high immobilization costs. To overcome these constraints, the maintenance policy has to ensure the vehicle availability and autonomy on given operating periods. The maintenance operations will then be performed before or after these periods.

Finally, to reduce the maintenance impact on the operating cost, the maintenance policy has to be optimized but it is also essential to lead a reflection linking both the system and its maintenance policy from the design stage. It requires the development of a methodology to guide the designer in his choices.

A maintenance policy based on the Maintenance Free Operating Period (MFOP) concept has been developed to ensure a high operating availability of the vehicle, assuming that maintenance actions are impossible during operating periods [97]. Otherwise, there will be significant additional costs. The MFOP defines an operating period without stops due to unplanned maintenance actions. This definition does not forbid failures as long as it does not affect the mission completion. It virtually differs all the future corrective maintenance operations to scheduled preventive maintenance periods. This feature enables to maximize the operating availability of the vehicle and reduce the uncertainty on the maintenance operations scheduling.

The developed MFOP-based maintenance policy aims at ensuring the proper functioning of a multi-component system on a given period with a specified confidence level. This policy is dynamic and is adjusted according to the available monitoring information on the components health state. It is composed of two stages:

- The first stage of the decision making process consists in defining at time t the need to have a maintenance operation on the system. This time corresponds either to the end of a MFOP or a breakdown on the system. To make the decision, it is necessary to estimate the probability that the systems operates until the end of the next MFOP knowing the available information at t . It is assumed that the information related to the multi-component system functioning is known at least at time t . If this system is operating, this probability corresponds to the Maintenance Free Operating Period Survivability (MFOPS) at time t . The MFOPS defines the probability that the system survives during the MFOP knowing that the system was operating at the beginning of the period. The available information on the health state can be integrated when computing the MFOPS. On the contrary, if the system fails at t , $MFOPS(t)$ is equal to 0.
- The second stage compares the $MFOPS(t)$ of the system with a specified confidence level NC . If $MFOPS(t) > NC$, the maintenance intervention is not necessary and the system can be deployed on the next MFOP without going to the workshop. Otherwise, if $MFOPS(t) < NC$, the maintenance intervention is essential.

As mentioned in the first stage, information related to the components health state is available. However, there are different levels of information. Either we have none (Level 1), or we have just an information if the component is operating or has failed (level 2), or we have access to a deterioration measurement (level 3). That information can have an impact on the calculation of the MFOPS:

- For the level 1, it is considered that no deterioration information is explicitly available.
- For level 2, the component functioning knowledge is available. The impact of that information is related to the system structure. If we have a series-structured system, that information does not influence the calculation of the MFOPS insofar as an operating series-system means that all the components are operating. However, for a component in a parallel structure, the operating/failed information affects the MFOPS calculation.
- For level 3, a deterioration measurement is available. This measurement describes more accurately the real state of the component than the two other levels and has a clear influence on the way to compute the MFOPS regardless of the system structure.

The usage conditions are also taken into account by considering two types of environments: a « normal » one and a « stressed » one. A mixture model is applied as we do not know when the component is evolving in one environment or in the other and when the environment changes occur.

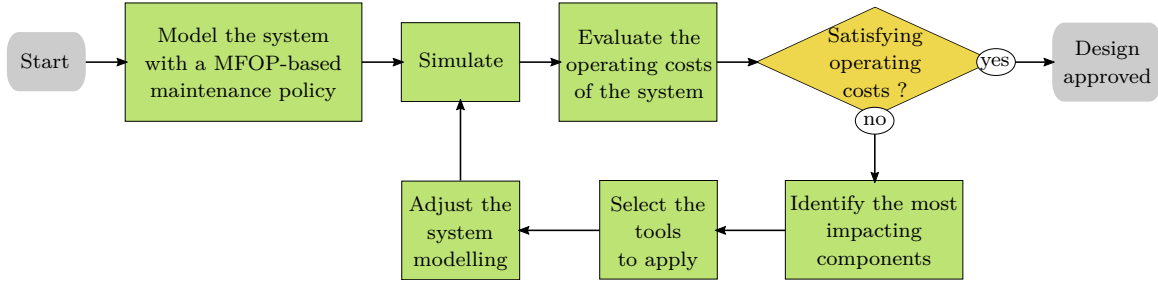


Figure 1.5: Design methodology for maintenance

Naturally, when a maintenance operation is required, the components to replace have to be identified to ensure the new MFOP with the required confidence level. To solve this constrained optimization problem, a genetic algorithm is applied to optimize a criterion based on the replacement costs and on the global efficiency of the maintenance operation. The criterion also takes into account the dependencies between the components.

As the customers place increasing emphasis on the systems sustainability and their associated services, it is necessary to lead a reflection on the costs of the systems during its whole lifetime. It also implies to consider the problems related to maintenance at the system design. A methodology has then been developed to optimize the design of a multi-component system in a MFOP perspective to reduce its operating costs [97]. This method integrates the MFOP-based maintenance policy described previously to place the issues related to the maintenance activity in the first phases of the system lifetime. Figure 1.5 describes the different steps of the methodology. The first step is to model the system structure by considering the dependences between the components and identifying their reliability models. The dynamic MFOP-based maintenance policy is also modelled in this phase. Then, a simulation phase based on the Monte Carlo method is necessary to estimate the system operating costs. This evaluation aims at making trade-offs between the following design proposals. According to the obtained operating costs, the system design is validated. If the costs are not satisfying, it is necessary to identify the system weak points. These are the components having the biggest impact on the operating costs. An importance factor is then built to identify and rank the components impact on the operating costs. After determining the weak points, the last stage consists in proposing design solutions to reduce their impact on the operating costs. These potential solutions are:

- Managing the components reliability features,
- Developing the monitoring architecture,
- Adding components redundancies,
- Managing components accessibility.

This methodology is considered as a decision support tool to make design recommendations.

The development of an MFOP-based maintenance policy and a design methodology for maintenance enables to ensure the vehicle operating availability on a given period and avoid breakdowns during the missions execution. However, some limits in the approach to design maintenance policies can be identified. The first one is that the operating periods, when the vehicle is performing missions, are assumed to be fixed. This hypothesis prevents us from taking into account the variability of the hauler activity. This period then has to adjust to the vehicle available periods and to the potential usage changes among the same period. Moreover, the suggested maintenance policies do not consider the opportunities offered by a fleet of systems. The capacity to integrate this « fleet » dimension is essential to optimize the operating costs, the customers productivity and to offer them customized services adapted to their needs.

1.3 Thesis contribution

The research work led through this PhD is the result of a collaboration between the Volvo Group and the lab GIPSA-Lab, belonging to the CNRS, Université Grenoble-Alpes and Grenoble INP. This collaboration aims at offering new solutions to solve the problems raised by the aftermarket department. These issues are a mix between maintenance scheduling problems and fleet management to consider the customer operating constraints.

This part firstly presents the industrial objectives to follow. Then, we align this joint problem between maintenance scheduling and fleet management with the existing contributions provided by the literature to identify the methodological challenges we will face to tackle this issue.

1.3.1 Industrial objectives

The Volvo Group wants to propose to its customers innovative services to maximize their productivity while fitting to their constraints. This offer aims at ensuring that the vehicles belonging to the fleet have a high availability level while reducing the impact of the maintenance costs on the operating income.

The objective is then to develop methods that enable to jointly optimize the maintenance operations scheduling and the fleet management. There is a double advantage for the customer. The maintenance operations will be scheduled without impacting the operating availability of the vehicles and the productivity will be increased by using at best the fleet capacity to dispatch the work between the different vehicles. This approach implies that we must no longer think only at the level of a vehicle but at the level of a fleet. The assignment of the missions to the different vehicles of the hauler will be based on the whole available dynamic information from the vehicles health state to the operating constraints. We can for instance cite as available information:

- the operating constraints for the customer,
- the reliability properties of the components,
- the monitoring information related to the health state of some components,
- the monitoring information related to the usage conditions of the vehicles,
- the features of the upcoming missions.

Before detailing the objective further, we have to clarify the definition of a mission. For the customer, a mission is a journey from the warehouse where the goods are loaded to one or several delivery points. In our case, we assume that a mission is a time period during which the usage conditions do not change. The vehicle then evolves in the same severity usage condition. It helps to configure the deterioration impact on the vehicle to traduce the usage severity. The mission enables to earn money when it is completed. But delay costs can be charged if the delivery deadline is not respected. To sum up, a mission is defined by:

- its duration,
- two parameters to define the deterioration model,
- the gain earned when the mission is completed without delay,
- the unitary delay cost charged for each time unit delay,
- its deadline.

The originality of this topic requires to develop a model to describe the behaviour and the health state of the vehicles in the fleet, to create a maintenance policy adapted to each vehicle, to manage a mission schedule and to implement a process to evaluate the model. This study can be decomposed into two big areas of focus.

The first one is to develop a method to jointly optimize the maintenance operations and the missions schedule considering the required usage conditions of the missions and their impact on the health state evolution. Naturally, we would like to schedule the maintenance operations during the inactivity phases of the vehicle. Contrary to the previous research work led by the Volvo Group [97], we consider that the missions durations are variable to meet the activity constraints of the hauler. After each mission, the policy will have to consider the available information to make a maintenance decision. This decision will integrate the monitoring information, the reliability features of the vehicle and the missions to complete on a given time horizon. The mission list can also evolve over time according to the deliveries required by the customer. To develop the maintenance policy, a maintenance model is necessary to integrate the operating and non-operating behaviour of the vehicle as well as the environment and operating conditions in which the vehicle evolves. Different tough points can be pointed in this development axis. The first difficulty lies in our capacity to identify the relevant variables to integrate when making the maintenance decision. The second one is to find the

adequate optimization method and the right decision criterion to manage both missions and maintenance operations scheduling.

The second development axis deals with fleet management. The possibilities offered by the fleet have to be integrated into the maintenance decision. The idea here is to have a maintenance policy adapted not just to the available information on a unique vehicle but to several vehicles available in a fleet. The sought advantage is to offer the opportunity to manage the assignment of the vehicles according to their health state, their capacity and features, the missions programs to complete and the available maintenance time slots. The missions order is not fixed at the beginning and new missions can be added during the schedule completion. Some missions can also have a higher priority than the others. It is an ambitious modelling program. The major difficulties lie in the choice of the optimization method to assign the vehicles to the different missions and more precisely in the decision criterion. Considering the fleet dimension to schedule the maintenance operations and the missions for the whole fleet adds a complexity degree with respect to the joint scheduling of missions and maintenance operations for a single vehicle.

1.3.2 Methodological challenges and thesis contribution

To tackle the industrial issues described in [Section 1.3.1](#), we are interested in the existing work to jointly schedule both missions and maintenance operations for a fleet of systems in the literature. The objective is to define a maintenance model adapted to the variations of usage conditions due to the different missions a system has to complete.

Some research works show a interest in the link between maintenance and production planning. The large majority of them is interested in scheduling the activities and the maintenance operations for machines and production workshops but few of them consider fleet of independent systems operating at the same time in different usage conditions.

Most of the time, the joint scheduling of maintenance and production is led for a single system while considering that all the production activities to do are known. In this context, optimization strategies are applied to solve this static joint scheduling problem. These strategies differ when it comes to the method itself but also regarding the chosen work hypotheses. Either the scheduling of both activities is sequential i.e. one activity, maintenance or production, is scheduled first and is used as an unavailability constraint to schedule the second activity [\[73\]](#). Or, an integrated strategy is applied that schedules simultaneously maintenance and production activities [\[57\]](#). The results of both methods are a unique schedule for both activities but the sequential strategy assumes that one activity has a higher priority than the other. The variety of approaches also comes from the chosen hypotheses in terms of optimization criterion, maintenance policy and deterioration model. The optimization criteria are either cost-related, by considering only maintenance costs [\[87\]](#) or sometimes maintenance and production costs [\[57\]](#), or time-related, when considering the completion times to complete the different tasks [\[35\]](#). Multi-objective optimization is also applied to optimize both maintenance and production costs as well as the completion time [\[46\]](#). The preventive maintenance

strategy also differs from one approach to another because we consider either perfect [35] or imperfect [57] maintenance. Finally, the models considering the effect of the system deterioration vary. It can affect the age of the system, its health state. In some cases, the models used are survival models that do not consider the possibility to record and integrate information related to the system deterioration evolution. When the deterioration is considered, it is within a deterministic framework in which the deterioration induced by each task is exactly known and no accidental failure can occur during the time horizon [87]. Few contributions treat the joint scheduling problem by considering random deterioration processes to integrate some uncertainty in the way to schedule the maintenance operations and the tasks.

In addition, the possibility to update the schedule and reschedule the remaining tasks and maintenance operations, according to real time information or events, is never considered. Indeed, rescheduling is a challenge because it is necessary to estimate if a rescheduling is worth it or if a schedule update will lead to expensive expenses in terms of resources and changes. The value brought by a piece of information has to be considered. When defining the scope of a rescheduling, three axes have to be taken into account: the rescheduling environment, the strategy and the methods [170]. The rescheduling problem is mostly studied without considering the maintenance activities. When they do consider maintenance, they only use simple maintenance strategy such as the schedule of only one maintenance slot [173].

The last point is that few research studies consider fleet of deteriorating systems to schedule maintenance and tasks. The fleet dimension is mostly covers when studying the job-shop, flow shop or parallel machines problems, but also for aircrafts. The traditional studies consider a limited number of objectives such as minimizing the sum of the completion times or minimizing the makespan [132]. However, in these models, downtime due to maintenance and system failure are generally formulated as a constraint. More modern studies start to integrate costs incurred by these activities as an objective to highlight the importance of intervention activities.

Actually, many research works study the joint scheduling problem but they just study a limited part of the global issue. There is no contribution that deals with the dynamic joint scheduling problem for a fleet of systems. Either they focus on the static scheduling part for a single system or a fleet of system, or they treat the rescheduling part but for a single system, and with simple maintenance policy or even just for production.hence, we have no complete and concrete methodology to tackle the whole dynamic joint scheduling issue for a fleet of stochastically deteriorating systems.

That is why, the methodological challenges lie in three areas: the way to jointly schedule maintenance and missions for a deteriorating vehicle, the way to integrate the available monitoring information and events to update the schedule and obtain a dynamic schedule, and how to integrate the fleet dimension when dynamically scheduling both activities. These three areas make up the three stages of the methodology to solve the complete joint scheduling problem for a fleet of vehicles. They enable to break down the whole problem into three smaller problems whose degree of complexity is gradual. Moreover, each stage works as a contribution in itself, but also defines the basis to develop the following stages.

Naturally, to jointly schedule maintenance and missions, an adapted maintenance policy has to be defined to consider the operating conditions changes and their impact on the vehicle deterioration. A reflection on the way to model the deterioration is also necessary to develop the maintenance model, but also to study which monitoring information can be retrieved, and to analyse the value offered by that information.

Note that we do not take into account any « geographical » or « spatial » aspects when studying the joint scheduling problem. Hence, the distance to travel during the missions, the distance to go to the maintenance workshop and the geographic location of the trucks are not considered.

1.4 Thesis outline

To solve these maintenance and missions scheduling problems, it is necessary to rely on a stochastic modelling based on the reality to convey the complexity of these issues. Chapters 2 and 3 introduce the elements necessary to build a maintenance model and Chapter 4 introduce the existing research work and methods to integrate maintenance and delivery tasks in the same schedule.

[Chapter 2](#) deals with the different ways to model the reliability of a system. It is the first element to build a maintenance model. After a few reminders on the reliability and its major features, this chapter presents the methods to model the failure through the physical, survival and deterioration models. They can be seen as three possible models of the same system but when having different information levels.

[Chapter 3](#) discusses the existing maintenance policies and the way to define them. Indeed, the maintenance policy is the second needed element to define a maintenance model. In the first place, the general notions related to maintenance are presented. We then focus on the use of monitoring information for the maintenance policy. Finally, a state of the art regarding the existing maintenance policies and the associated models for mono and multi-component systems is led. It is essential to model maintenance to evaluate the performance and optimize the maintenance schedule.

[Chapter 4](#) displays a state of the art regarding the existing contributions to jointly schedule maintenance and production for a single system and for a fleet of systems. To complete it, a study regarding the methodology to consider monitoring information or the occurring events as opportunities for rescheduling is led. We will conclude this chapter by showing that the existing approaches do not adequately address the problem as we define it, i.e. with systems that stochastically deteriorate. Finally, a focus point is made on one of the optimization method, the genetic algorithm, insofar as it is the core method that will be used in the next chapters.

Based on the contents presented in these chapters, the three-step methodology can be described through the three following chapters.

[Chapter 5](#) details the methodology applied to jointly schedule missions and maintenance for a single deteriorating vehicle in the static case. We introduce the static joint scheduling problem and present the genetic algorithm based resolution approach. An application example is also led to analyse the performance of the methodology and its sensitivity as well as to characterize the limits of this approach.

[Chapter 6](#) is focused on the dynamic aspect of the joint scheduling problem for one vehicle and describes how monitoring information related to the vehicle health state, failure occurrences and new available missions can be integrated in the decision-making process to update the schedule. It completes the approach developed in [Chapter 5](#). The methodology is described and evaluated through application examples to study the value of the information brought by the monitoring systems, but also to show the interest in updating the schedule when the hauler activity evolves.

In [Chapter 7](#), we focus on the integration of the fleet dimension to schedule the missions and the maintenance operations for the whole fleet of vehicles. This part is an improvement of [Chapter 5](#) and [Chapter 6](#) insofar as it integrates the fleet dimension in the static and the dynamic cases. It also studies the possibility to have either an homogeneous or an heterogeneous fleet of vehicles. The deterioration evolution on the same mission can then vary if the vehicles have different configurations. This chapter is structured in the same way as [Chapter 5](#) and [Chapter 6](#) with a description of the methodology and application examples to illustrate the advantages and the limits of such method.

[Chapter 8](#) concludes our study and proposes perspectives raised by our research work.

Reliability modelling

Contents

2.1 Reliability reminders	24
2.1.1 Reliability and dependability	24
2.1.2 The major features in reliability	25
2.1.3 Reliability estimation methods	28
2.2 Physical models	28
2.3 Survival models	29
2.3.1 Impact of the environment on the survival models	30
2.4 Deterioration models	31
2.4.1 Deterioration classes	32
2.4.1.1 The discrete deterioration models	32
2.4.1.2 The continuous deterioration models	33
2.4.2 Levy processes	34
2.4.2.1 The Wiener process	34
2.4.2.2 The Gamma process	34
2.4.3 Impact of the operating environment on the deterioration model	36
2.5 Conclusion	37

A failure occurs when a component cannot ensure its required function anymore and leads to a system breakdown. Maintenance aims at preventing these failures and taking action on the system to restore its functionalities after a breakdown. Its principal objective is then to maintain and improve the system reliability.

Being able to model the failure and the system deterioration becomes a key element to optimize the maintenance policy.

This chapter proposes a definition of the reliability and its usual features. Then, a focus is made on the stochastic modelling of the component deterioration. Deterioration models quality depends a lot on the available monitoring information regarding the operating environment and the usage conditions.

2.1 Reliability reminders

This part is a general overview of the concepts associated with reliability and the existing methods that can be used in the industry to evaluate the reliability of a system.

2.1.1 Reliability and dependability

Dependability represents the science of failure and breakdown. The objective is to build systems for which the failures are anticipated and tolerated. As mentioned by Laprie [89], the dependability of a system is the property that enables users of the system to place justified trust in the service it provides to them.

Definition 1: *Dependability refers to all the skills of an asset that enable it to perform a required function at the appropriate time, for the expected duration, without damage to itself and its environment.*

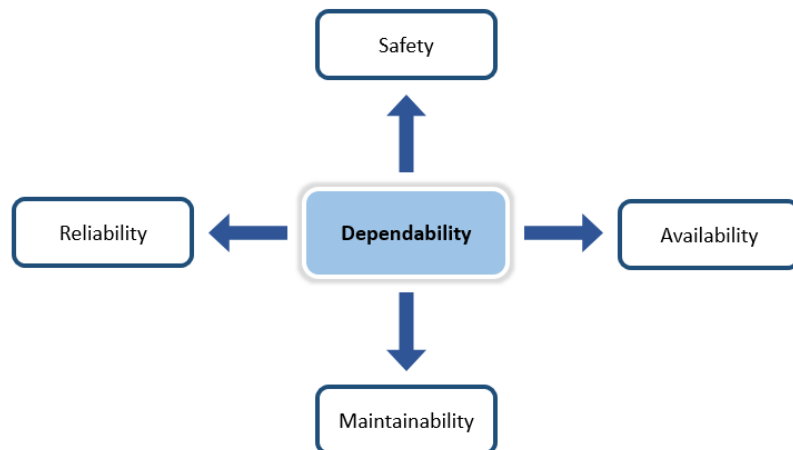


Figure 2.1: The concepts associated with dependability

Dependability is composed of four concepts: reliability, availability, maintainability and safety (Figure 2.1).

Definition 2: *The safety of an entity is defined as its ability to avoid the occurrence, under given conditions, of critical or catastrophic events.*

This component is essential to consider for systems for which the risk of damage for users and for the environment is high.

Definition 3: *The availability of an entity is defined as its ability to perform a required function under given conditions at a given time, assuming that the provision of necessary external resources is ensured [3].*

When a problem regarding either the safety or the availability of the entity is detected, the notion of reliability is considered.

Definition 4: *The reliability is the ability of an entity to perform a required function or to satisfy user needs under given conditions, for a given period of time [3].*

The probability that the entity performs its function on a given time period enables to measure the entity reliability. When a failure appears on the entity that needs an action to restore its functionality, the notion of maintainability is introduced.

Definition 5: *The maintainability of an entity is defined as its ability to be maintained or restored, over a given time interval, in a state in which it can perform a required function. This implies that maintenance is performed under given conditions, with prescribed procedures and means [3].*

2.1.2 The major features in reliability

In this sub-part, a probabilistic definition of the reliability and its characteristics is given [20], [137]. The study of reliability uses a priori knowledge and the statistic treatment of data to estimate the failure probability of an entity during its lifetime.

Let us consider T a random value representing the functioning duration of an entity before observing the failure. This variable is supposed positive and continuous.

The distribution function $F(t)$ of the random variable T is given by Eq. 2.1.

$$\forall t \geq 0, \quad F(t) = P(T \leq t) \quad (2.1)$$

This function represents the probability that the entity has a breakdown in the time interval $[0; t]$. This function has different properties (Eq. 2.2).

$$\lim_{t \rightarrow 0} F(t) = 0 \quad \text{and} \quad \lim_{t \rightarrow \infty} F(t) = 1 \quad (2.2)$$

The probability density $f(t)$ of the random variable T is defined in (Eq. 2.3).

$$f(t) = \frac{dF(t)}{dt} = \lim_{\Delta t \rightarrow 0} \frac{P(t < T \leq t + \Delta t)}{\Delta t} \quad (2.3)$$

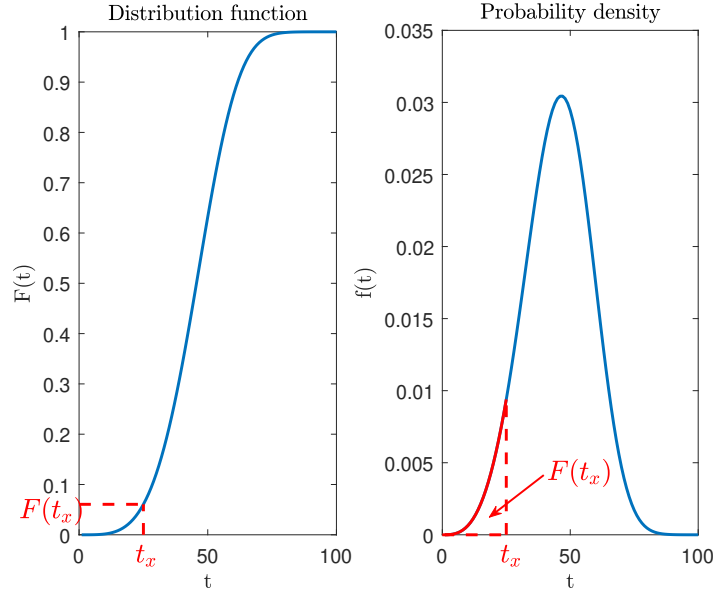


Figure 2.2: Distribution function and probability density

When Δt is small, $f(t) \times \Delta t$ represents the probability to have a failure on a small time interval after the time t . The distribution function $F(t)$ and the probability density $f(t)$ are represented in Figure 2.2. The distribution function $F(t_x)$ corresponds to the area under the probability density curve between 0 and t_x .

The reliability $R(t)$ (Eq. 2.4) is the probability that the entity does not have a breakdown in the time interval $[0; t]$. In other words, it is the probability that the entity survives during the time period $[0; t]$ and still works at t .

$$\forall t \geq 0, \quad R(t) = 1 - F(t) = P(T > t) = 1 - \int_0^t f(u) du \quad (2.4)$$

The conditional probability is the probability that the entity has a breakdown during the time period $[t; t + \delta t]$ knowing that it was still functioning at t (Eq. 2.5).

$$P(t < T \leq t + \Delta t | T > t) = \frac{P(t < T \leq t + \Delta t)}{P(T > t)} = \frac{F(t + \Delta t) - F(t)}{R(t)} \quad (2.5)$$

To obtain the failure rate or hazard rate $z(t)$, we just have to divide the conditional probability by the length of the time interval Δt and then make Δt tend towards 0 (Eq. 2.6).

$$\forall t \geq 0, \quad z(t) = \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} \times \frac{P(t < T \leq t + \Delta t)}{P(T > t)} = \frac{f(t)}{R(t)} \quad (2.6)$$

The failure rate provides information on the wear or ageing of the entity. When the failure rate is growing with time, the entity is ageing. Ageing means in this case that the failure conditional probability for the entity is increasing with time. On the contrary, if the failure rate decreases with time, this conditional probability decreases as time passes. The entity is then getting younger.

In general terms, the behaviour of the failure rate for a non-repairable entity looks like a bathtub curve (Figure 2.3). The failure rate is composed of three parts:

- During the burn-in period (1), the failure rate is firstly high and then gradually decreases. It corresponds to early failures observable on a new entity (burn-in, elimination of youth defects). This period is known as the infant mortality period.
- During the useful life period (2), the failure rate is constant. It represents the most important phase in the entity life because it is the moment when the number of failures is the smallest.
- During the wear out period (3), the failure rate increases because the entity is ageing. The entity wears out due to the accumulation of shocks, fatigue and so on.

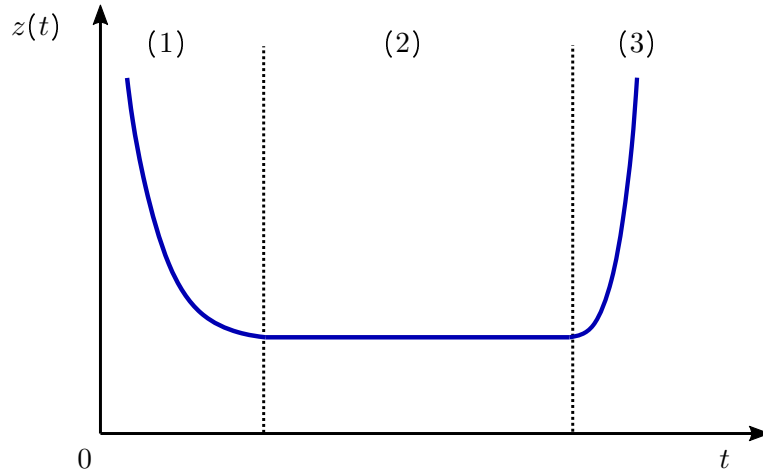


Figure 2.3: General behaviour of the failure rate

The last major feature used to characterize the reliability of an item is the Mean Time To Failure, also known as the MTTF (Eq. 2.7). It is the average duration during which the entity is well operating before its failure. The MTTF can be assimilated to the integral of the reliability if it is a finite quantity (Eq. 2.8).

$$MTTF = \mathbb{E}[T] = \int_0^{\infty} t f(t) dt \quad (2.7)$$

$$\text{If } MTTF < \infty, \quad MTTF = \int_0^{\infty} R(t) dt \quad (2.8)$$

This indicator is often used in the industry to compare the reliabilities of items provided by different manufacturers. However, it is not relevant to consider it when it comes to make maintenance decisions.

2.1.3 Reliability estimation methods

To estimate the reliability of an item in a practical way, three principal methods can be applied.

- The first method calls for the use of generic reliability databases such as MIL-HDBK217F [114] and OREDA [128]. Both contain failure rate estimates for specific components. However, the military handbook MIL-HDBK217F only contains failure rate estimates related to primary failures. On the contrary, the Offshore Reliability Data (OREDA) handbooks contains failure rate estimates for both primary failures and common cause failures.
- The second method consists in running trials on the item to estimate its reliability. These trials aim at placing the entity in operating conditions close to the reality to check its functioning. However, in the industry, many components have a very long lifetime. It is then unimaginable for a company to run such long trials due to time and cost constraints. A common way to tackle this problem is to expose the item to sufficient overstress to bring the MTTF down to an acceptable level. Then, one extrapolates from the obtained information under stressed conditions to normal use conditions. This approach is called accelerated life testing [121]. The most complex part in these trials is to accurately reproduce the entities actual operating constraints.
- Finally, the last method consists in estimating the reliability of an entity based on the data collected from experience feedbacks. The Volvo Group uses the vehicle maintenance history to estimate the reliability of the different components. The advantage is that we focus on the reliability as it is observed for the vehicles users. However, to have good quality reliability models, it is necessary to wait for several years to have enough usable data.

To use the experience feedbacks, statistical methods can be applied to process the data and to suggest realistic models to represent the reliability variables defined in Section 2.1.2. These models will be presented in the following sections. There are either physical models (see Section 2.2) or stochastic models (see Section 2.3, Section 2.4).

2.2 Physical models

The physical models, called "white box" models [123], are analytical models describing the deterioration process of an entity. For instance, the cumulative damage is modelled by the

fatigue cycles of an entity and directly linked with the failure through a mathematical expression [67]. However, this kind of models is very hard to build because it has to formalize the relationships between several mechanical and physical features characterizing both the entity and its environment [31].

2.3 Survival models

Survival models enable to have a stochastic modelling of the failure. They are considered as "black box" models [123].

For these models, the only available information regarding the entity is binary (Figure 2.4) and indicates whether the entity is working or if it is down. Its state at time t can be described by a state variable $X(t)$ (Eq. 2.9).

$$X(t) = \begin{cases} 1 & \text{if the entity is working at time } t \\ 0 & \text{if the entity is down at time } t \end{cases} \quad (2.9)$$

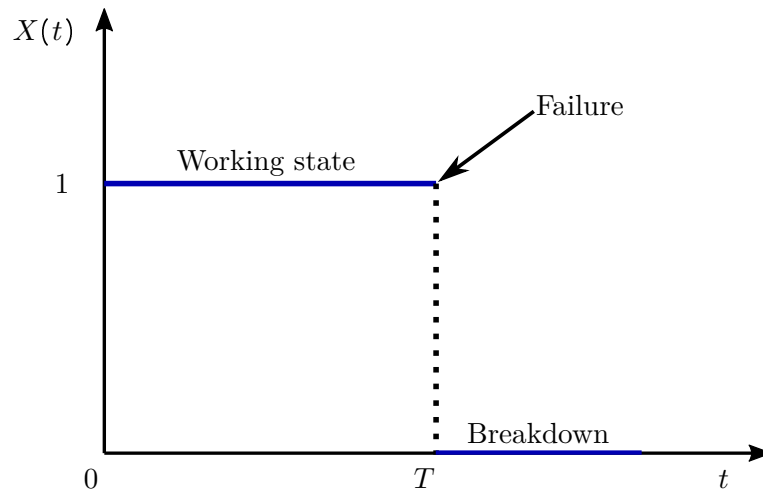


Figure 2.4: Information about the entity operating state

The intermediate states between the "new" state and the failure are not considered. The objective of the survival models is to define the probability law describing the operating time T of the entity. It enables to link the time with the failure occurrence through the failure rate $z(t)$.

To estimate the law parameters and the failure rate of the entity, trials can be carried out or the data collected from experience feedbacks [137] can be used (see Section 2.1.3).

Let us consider a time interval $[0; t]$ sampled in p disjointed intervals noted τ_i . These intervals have the same length such as $\tau_i = i \cdot \Delta t$ with $i = \{1, \dots, p\}$. We denote:

- $N_{X=0}(i)$ the number of entities that failed during the interval τ_i .
- $N_{X=1}(i)$ the number of entities working at the beginning of the interval τ_i .

The experience is carried out on N_t entities ($N_{X=1}(1) = N_t$) and the failure rate can be estimated as in Eq. 2.10.

$$z(\tau_i) \approx \frac{N_{X=0}(i)}{N_{X=1}(i)\Delta t} \quad (2.10)$$

The recorded information can be censored for different reasons:

- The duration of the experience is shorter than the entity failure occurrence time.
- The experience stops once the number of failures reaches a predetermined value.
- The two previous censoring type are combined.

This information loss leads to the development of estimation methods for the survival probabilities such as Nelson or Kaplan-Meier non parametric estimators [137].

A survival model only gives a binary information about the functioning state of the entity and supposes that an entity, in a functioning state at time t , fulfils its mission as well as another entity also operating at time t . Yet, in practice, it is unlikely that two same entities operate with exactly the same health state at the same time. This kind of model does not enable to individually follow a specific entity. To estimate the health state of an entity, it is necessary to use a deterioration model (see Section 2.4).

2.3.1 Impact of the environment on the survival models

A commercial heavy vehicle is a complex system to model because it evolves in a very changing operating environment. The characteristics of this dynamic environment can change the nature of the deterioration phenomenon as well as its speed. To this extend, it is not possible to neglect the impact of the environment on the components' deterioration. We have to be able to model the effect of this dynamic environment on the component's progress towards the failure.

The literature shows a growing interest for taking into account the covariates [155]. A covariate is an external variable that influences the deterioration of the entity. The effect of the covariates on the deterioration vary from one to the other. It is then necessary to identify the most representative covariates to model at best the deterioration phenomenon. This identification can often be made with the help of components experts. For the industrial systems, these covariates are often related to the usage conditions and the operating environment [80]. Most of the time, for an industrial vehicle, these covariates are:

- the road topography,
- the elevation,
- the outside temperature,
- the vehicle load,
- the kind of trip,
- the road surface,
- and so on.

If we consider a specific component such as the brake pads, other parameters like the vehicle average speed, the average fuel consumption, the location of the pad, the braking pressure are also identified as covariates [31].

They are different survival models integrating covariates [65], [118]. One of the most used model is the proportional hazards model. It enables to express the failure rate according to time and the explanatory variables. The hypothesis of the proportional risks means that the risk ratio between an entity having a given feature and another one not having this specific feature is constant at every time t . In other words, the risk of an entity having a given feature is multiplied by a constant, compared to the entities that do not have this feature [49], [97]. Among this kind of models, the Cox model [45] is the most known.

Taking the covariates into account to build the survival model enables to improve its quality as we consider the dynamic environment. However, these models are still unable to report on the health state of a specific component. Deterioration models are then necessary to model the health state evolution of a component (see Section 2.4).

2.4 Deterioration models

The survival models presented in Section 2.3 do not enable to describe the evolution of an entity towards the breakdown. To model this evolution in an accurate way, a second class of models, called deterioration models, is introduced. These "grey box" models [123] describe an observable variable to quantify the entities health state rather than describing through physical laws all the mechanisms causing the deterioration, or limiting ourselves to survival models.

Contrary to the survival models, the state space of the deterioration models is not limited to an operating state and a breakdown state. We suppose that there are some intermediate states, called deteriorated states. To apply these models, the first stage consists in defining a deterioration indicator directly linked with the deterioration state of the entity. The principal objective aims at representing the evolution of this indicator, denoted $Z(t)$, also called the deterioration trajectory. Figure 2.5 represents the deterioration trajectory of an entity. This

entity is down at t_L . It is the moment when the trajectory exceeds a fixed wear threshold L . To some extent, survival and deterioration models can be seen as two possible models of a same entity with different information level.

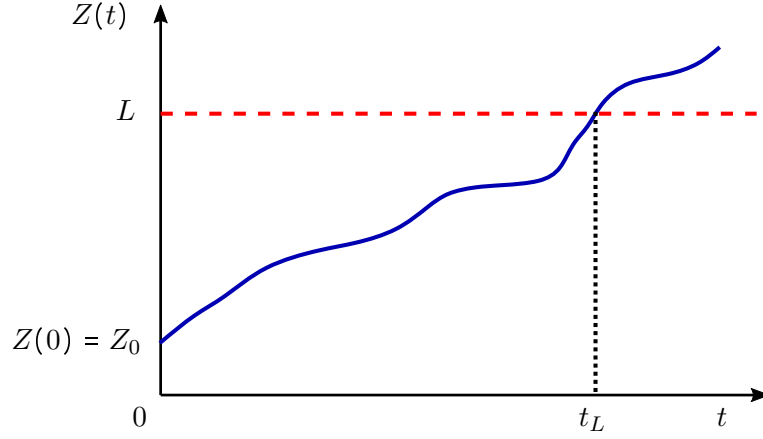


Figure 2.5: Deterioration trajectory of an entity

This part starts with a presentation of the different existing models to describe the evolution of the deterioration indicator. The model choice is based on the nature of the deterioration we try to model. Then, we focus on a specific model, the Gamma process, very used to represent the deterioration of industrial systems. Finally, we introduce the way to integrate knowledge regarding the dynamic environment into the deterioration modelling.

2.4.1 Deterioration classes

As mentioned in the previous part, the deterioration models consider intermediate states between the initial operating state and the failure. To model the deterioration evolution for an entity, it is necessary to identify the transition laws between the different states. This modelling considers the nature of the deterioration indicator that we want to follow. In the literature, the deterioration models are separated into two classes: the discrete deterioration models and the continuous deterioration models [49], [97].

2.4.1.1 The discrete deterioration models

The discrete deterioration models refer to shock-type deteriorations. Each shock, whose occurrence is random, increases the deterioration of the entity. This deterioration increment can be deterministic or random. A shock causes a jump from one deterioration state to another (Figure 2.6). Between two consecutive shocks, the deterioration level stays the same. The deterioration increments add up until the total failure of the entity. In this case, Markov or semi-Markov processes can be used to model both the moment of the shock and the state in which the entity is [36], [55]. In practice, it can be difficult to characterize the transitions between the different states. Moreover, we need a high amount of data to ensure the estimation

relevance.

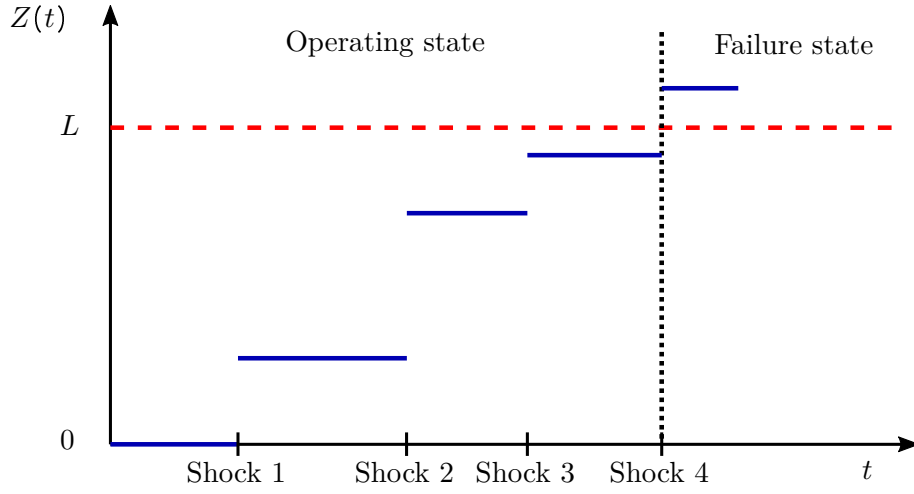


Figure 2.6: Representation of a discrete deterioration model

2.4.1.2 The continuous deterioration models

The continuous deterioration models are more relevant to model phenomena such as fatigue, gradual erosion of a material or gradual wear of a mechanical component [97]. For these models, it is necessary to know the law followed by the deterioration increments between two consecutive times to predict the deterioration level at every moment (Figure 2.7).

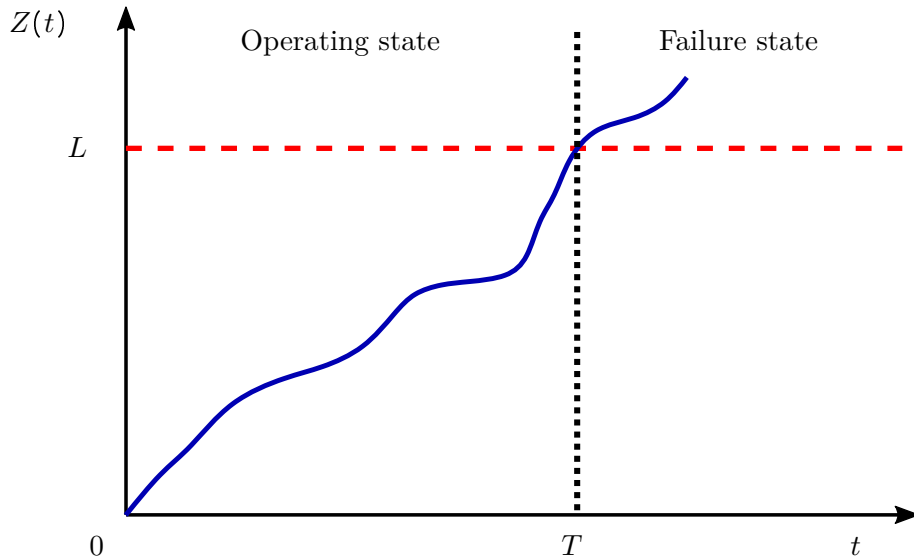


Figure 2.7: Representation of a continuous deterioration model

We generally consider that the deterioration process is a Markov process. This hypothesis is justified if the only available information on the entity health state is the growth of its

deterioration between two observations. The properties regarding the independence and the stationarity of the increments lead to the selection of the Levy processes class [13] to model an entity deterioration. These processes are detailed in the [Section 2.4.2](#).

2.4.2 Levy processes

The continuous deterioration models fit to represent observable phenomena on many industrial systems. The Levy processes in particular group processes that are widely studied in the literature [146]. The Levy processes family is composed of two main types of processes: the Wiener processes and the Gamma processes.

2.4.2.1 The Wiener process

The Wiener process is a Levy process with a continuous trajectory. Its deterioration increments follow normal laws. This process enables to model a continuous and increasing on average deterioration. However, the deterioration is not monotonous. It comes from the fact that the deterioration increment has a non-zero probability of being negative. This property can be tolerated in some conditions and work has been developed on the basis of this type of modelling [11], [180]. Nonetheless, we cannot model monotonous deteriorations such as crack propagation or abrasion wear phenomena. It is then necessary to choose processes that favour monotony.

2.4.2.2 The Gamma process

Theoretically, a Levy process cannot be used to model a deterioration that is both continuous and increasing. The solution consists in placing oneself in the limit case of a jump process with an infinite number of jumps over a finite time interval. To do so, the probability law followed by the deterioration increments is defined for positive values and is infinitely divisible. It leads us to introduce a process that respects these conditions: the Gamma process.

The Gamma process is a Levy process whose deterioration increments follow a Gamma law [1], [168]. This feature gives a high flexibility to model very different deterioration behaviours. Note that the Gamma law is defined for positive values. Therefore, the Gamma process defines a positive increment process. It is suitable for continuous and monotonous degradation modelling, unlike the Wiener process. We model the fact that the entity health state cannot be improved without external intervention. This specific process is widely used in the literature to model various phenomena like abrasion wear, fatigue, corrosion, crack growth, and so on [31], [97], [168].

A stationary Gamma process $X(t), t > 0$ is defined by two parameters: its shape parameter α and its scale parameter β . It respects the following properties:

- $X(0) = 0$;
- The increments of $X(t)$ are independent;
- $\forall t > 0$ and $\forall h > 0$, the growth law $X(t + h) - X(t)$ follows a Gamma law whose probability density is defined in Eq. 2.11.

$$f(x) = \frac{\beta^{\alpha h}}{\Gamma(\alpha h)} x^{\alpha h - 1} e^{-\beta x} \text{ for } x > 0 \quad (2.11)$$

With $\Gamma(\cdot)$ the Gamma function defined in Eq. 2.12.

$$\forall x > 0, \quad \Gamma(x) = \int_0^\infty t^{x-1} e^{-t} dt \quad (2.12)$$

To estimate the parameters of the Gamma process, the most used methods are the maximum likelihood method and the method of moments [168]. We can then define the expected value E and the variance V of the increments (Eq. 2.13).

$$E = \frac{\alpha}{\beta} \text{ and } V = \frac{\alpha}{\beta^2} \quad (2.13)$$

If we want to define the expected value and the variance of the Gamma process, we just have to multiply E and V by the time t . As we have a stationary Gamma process, the expected deterioration is linear in time.

Figure 2.8 shows an example of deterioration trajectories for a Gamma process.

For the entities whose deterioration is defined by a Gamma process, the failure occurs due to an excess of deterioration. It means that the indicator $Z(t) > L$, where L is the wear threshold. The distribution function and the reliability can then be respectively expressed as in Eq. 2.14 and Eq. 2.15.

$$F(t) = P(T \leq t) = \frac{\Gamma(\alpha t, L\beta)}{\Gamma(\alpha t)} \quad (2.14)$$

$$R(t) = P(T > t) = P(Z(t) < L) = 1 - \frac{\Gamma(\alpha t, L\beta)}{\Gamma(\alpha t)} \quad (2.15)$$

With $\Gamma(\alpha, L\beta)$ the incomplete Gamma function defined in Eq. 2.16.

$$\Gamma(\alpha t, L\beta) = \int_{L\beta}^\infty u^{\alpha t - 1} e^{-u} du \quad (2.16)$$

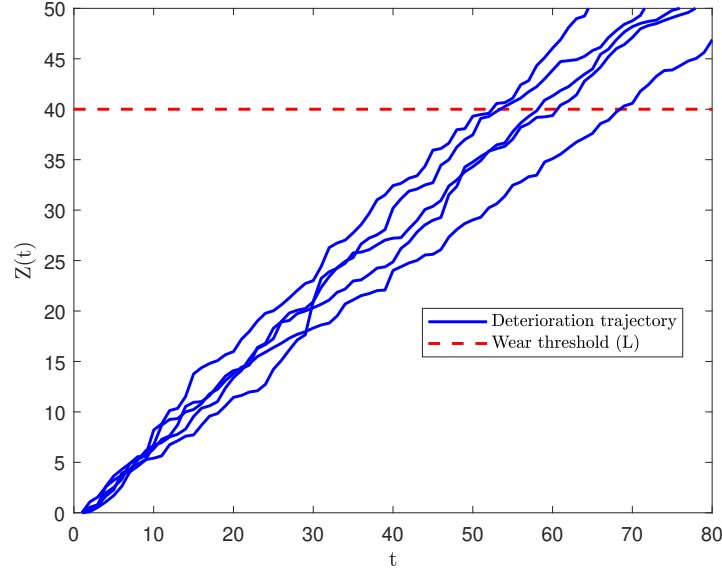


Figure 2.8: Deterioration trajectories from a Gamma process whose parameters are $\alpha = 2$ and $\beta = 3$

It is interesting to note that knowing the indicator $Z(t)$ enables to update the conditional reliability more accurately than with the simple knowledge of the operating state or failure state. The indicator information can change the remaining useful life estimation of the entity and possibly modify the maintenance decisions. These different points are detailed in [Chapter 3](#).

2.4.3 Impact of the operating environment on the deterioration model

In [Section 2.3.1](#), we explained that the dynamic environment in which the industrial vehicle evolves can have an impact on the components deterioration. The objective is then to build a deterioration model that considers the additional information provided by the covariates [\[64\]](#). It enables to better explain the deterioration phenomenon. The capacity to integrate the dynamical environment in the deterioration models aims at improving the maintenance decision-making process.

The covariates effect can take different forms. It can directly have an impact of the deterioration [\[50\]](#). In the first place, the impact can be occasional [\[178\]](#) by causing a sudden increase of the deterioration level. It can also be temporal and modify the speed and/ or the variance of the deterioration [\[190\]](#). In the case of the Gamma process, this temporal effect is taking into account through the shape parameter [\[16\]](#) and/or the scale parameter [\[90\]](#) that will be defined according to the covariates state.

2.5 Conclusion

To estimate the reliability features of an entity, there are different models. The physical models enable to represent the physical phenomena through mathematical expressions. However, these phenomena can be difficult to model for complex entities. The survival models, easier to set up, give a priori characteristics but do not enable to estimate the entity health state over time. The deterioration models allow us to estimate the future deterioration level according to an initial level and to compute the failure probability for a specific entity. The advantage of this model is that the reliability features can be updated each time a deterioration measurement is available.

To build a deterioration model, it is necessary to collect data on the entity. The deterioration process parameters can then be estimated based on the collected deterioration measurements and their recording time. However, to ensure a better model quality, it is relevant to also identify the entity usage conditions and monitor parameters correlated to the dynamic environment.

Our objective is to develop a dynamic method to schedule both the missions and the maintenance operations for a fleet of vehicles. The idea is then to use a deterioration model adapted to the vehicle usage and the changes in the operating conditions to make a prognosis about the vehicles health state. This prognosis will enable to make decision regarding the future activities the vehicle can do, i.e. another mission adapted to its health state or a maintenance operation. The predictive maintenance operations for each vehicle will then be decided by considering the residual life time estimations. The maintenance schedule for the fleet will then be adapted to the actual vehicles deterioration and to their usage conditions defined by the different missions profiles.

For the following chapters, each vehicle health state is estimated by using a global health indicator. It means that the deterioration model is defined by considering the vehicle as a unique component. This choice is justified by the fact that the major part of this work is focused on fleet management rather than on considering each vehicle as a multi-component system. Many research studies have already been done on vehicles modelling as multi-component systems [31], [97]. The information brought by the deterioration model are then integrated to the maintenance decision-making process. The next necessary step is to define a convenient maintenance model adapted to the deterioration information.

Maintenance modelling

Contents

3.1	General information about maintenance	40
3.1.1	Different types of maintenance	41
3.1.2	Effect of the maintenance actions on the systems	43
3.1.2.1	Perfect maintenance	43
3.1.2.2	Minimum maintenance	43
3.1.2.3	Imperfect maintenance	44
3.1.3	Maintenance model	44
3.1.4	Performance evaluation and optimization	44
3.2	Maintenance and monitoring systems	46
3.2.1	Monitoring information	47
3.2.1.1	Monitoring of the entity state	47
3.2.1.2	Monitoring of the environment	48
3.2.1.3	Monitoring cost	49
3.2.2	Impact of the monitoring information on the decision-making process	49
3.3	Maintenance policies for mono-component systems	51
3.3.1	Maintenance policies based on the system lifetime law	51
3.3.1.1	Age replacement policy	52
3.3.1.2	Block replacement policy	54
3.3.1.3	Inspection-based policy	56
3.3.2	Policies based on the current system state	57
3.3.2.1	Condition-based maintenance	58
3.3.2.2	Predictive maintenance policy based on the remaining useful life	59
3.4	Maintenance policies for multi-component systems	61
3.4.1	Static grouping policies	62
3.4.1.1	Corrective maintenance grouping	62
3.4.1.2	Grouping of scheduled preventive maintenance operations	62
3.4.1.3	Opportunistic maintenance	65
3.4.2	Dynamic grouping	65
3.5	Conclusion	67

Maintenance management is a real economic challenge for the companies. Indeed, this function enables to ensure that the industrial systems operate well but also shows a form of pro-activity to the customers on how to handle their vehicles from an aftermarket perspective.

This chapter aims at providing an overview regarding the existing maintenance policies and how they are defined, evaluated and optimized. The objective of a maintenance policy is to prevent, avoid or correct the malfunctions of a system. It consists in defining a set of rules to schedule the right maintenance operations at the right time. In the long term, the aim is to determine a maintenance policy capable of optimizing the defined criteria.

To tackle such issues, we need to build a model that can report on the health state evolution of a system subject to a specific maintenance policy. A maintenance model can be seen as a real decision-support tool used to evaluate and compare the performances of different maintenance policies.

In [Chapter 2](#), we focused on the modelling of a system reliability. This part was the first stage to build a maintenance model. This chapter presents the global notions associated with maintenance and brings a recap of the existing maintenance policies for mono and multi-component systems used in the literature. It enables to specify the context of our research work, especially how to define and optimize the maintenance decision-making process.

3.1 General information about maintenance

Definition 6: *Maintenance is all the technical, administrative and management actions made during the life cycle of an asset. They are intended to maintain or restore it in a state in which it can perform the required function [2].*

The maintenance function groups different actions that affect different research domains [29]: costs optimization, spare parts management, scheduling maintenance operations, skills necessary for the diagnosis, for the prediction to perform the preventive maintenance operations and so on. To schedule the maintenance operations for a system, it is necessary to fix some rules from which a maintenance decision can be made. These rules are grouped into a maintenance policy. The maintenance policy varies according to the type of maintenance actions we consider.

This definition introduces the two sides of maintenance. The first one, through the verb « maintain », is related to the notion of preventing for an entity that is still working. The second one associates the verb « restore » to the corrective aspect of maintenance. It means repairing an entity that cannot correctly perform its function anymore. Based on this definition, we will present the different types of maintenance and effects maintenance can have on systems as well as the ways to evaluate the performances.

3.1.1 Different types of maintenance

To classify the different types of maintenance, two main categories are defined: corrective maintenance and preventive maintenance [137]. These categories are themselves composed of some subcategories. This classification is presented in Figure 3.1.

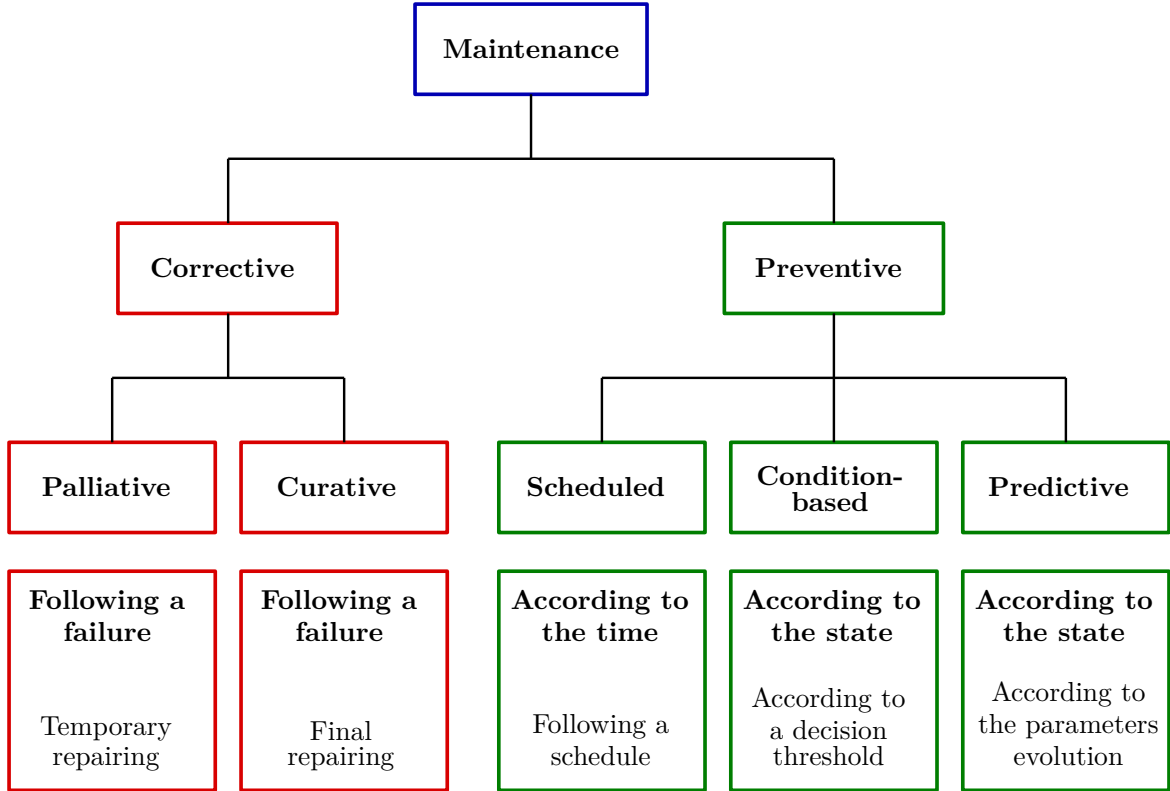


Figure 3.1: The different types of maintenance

Definition 7: *Corrective maintenance is the maintenance action performed after the detection of a failure. It aims at restoring an asset to a condition in which it can perform a required function [2].*

Corrective maintenance is often referred to as passive maintenance because it is the entity breakdown that leads to the maintenance action. It means that we are in a defensive position in the event of a failure. This category is made up of two subcategories: palliative maintenance and curative maintenance. Palliative maintenance deals with temporary troubleshooting operations allowing an entity to perform the whole required function or at least a part of it. On the contrary, curative maintenance describes repairs enabling the entity to return to its original state. In our work, we consider only curative maintenance operations as corrective maintenance.

Corrective maintenance makes sense when an entity breakdown does not have major economic consequences or when the constraints in terms of safety are low. However, for some industrial systems, waiting for the complete breakdown of the system can be critical and very

expensive [165]. To this extend, companies prefer to maintain preventively the system, i.e. intervene on it before the breakdown. This is the purpose of the second maintenance category.

Definition 8: *Preventive maintenance is the maintenance action performed at predetermined intervals or according to prescribed criteria. It aims at reducing the failure probability or deterioration of an asset functioning [2].*

Unlike corrective maintenance, acting on an entity before it fails is part of a proactive approach. Preventive maintenance is then of interest for different reasons:

- either for safety reasons when the failure consequences are deemed unacceptable,
- or for economic reasons when preventive maintenance is much more cost-effective than passively waiting for the failure,
- or for practical reasons requiring maintenance at specific times.

Preventive maintenance is divided into three subsections: scheduled maintenance, condition-based maintenance and predictive maintenance.

Definition 9: *Scheduled maintenance is a preventive maintenance action carried out at pre-established time interval or according to a defined number of units of use but without prior checking of the asset health state [2].*

With this maintenance policy, maintenance operations are carried out based on a fixed schedule. The periods between two operations are specified, either by a time interval or by operating cycles. The objective is only to determine the operations frequency to optimize the pre-defined criteria. This type of maintenance is easy to set up. It is also useful when the entity cannot be monitored or when monitoring it generates costs higher than the expected gains.

Definition 10: *Condition-based maintenance corresponds to preventive maintenance based on a monitoring of the asset functioning and/ or of significant parameters for this functioning, integrating the actions that result from it [2].*

This maintenance policy is subject to the appearance of some parameters revealing the deterioration or a performance decline of the concerned entity. The collected or measured data are compared to a predetermined threshold, also known as the decision threshold. Crossing the threshold acts as an alert that triggers a maintenance operation.

Definition 11: *Predictive maintenance corresponds to conditional maintenance carried out by following the extrapolated forecasts of the analysis and of the evaluation of significant parameters of the asset deterioration [2].*

It should be noted that predictive maintenance also includes the future operating conditions and environment in which the system will evolve. Unlike condition-based maintenance, it combines a forecast of the health indicators to make maintenance decisions. To consider

setting up predictive maintenance, it is necessary to control in detail the behaviour of the considered entity. The use of this knowledge makes it possible to anticipate and predict at best when the maintenance operation should be performed.

Predictive maintenance is probably the policy requiring the biggest investments regarding the other policies. This cost increase can be mainly explained by the monitoring technologies used to collect data on the system behaviour and on the usage conditions. It requires many sensors, software, and so on. In this context, the first analysis to carry out is a cost/gain analysis to evaluate if setting up this kind of maintenance policy is relevant for a specific entity.

These definitions set the framework when studying the maintenance field. However, in real life, a mixed maintenance policy is generally applied. It combines corrective and preventive maintenance. The optimal balance between corrective and preventive maintenance is obtained thanks to maintenance modelling.

3.1.2 Effect of the maintenance actions on the systems

After introducing the different types of maintenance, we will focus on the different way the maintenance can act on the entities. To do so, we will differentiate the maintenance operations, namely perfect, minimum and imperfect maintenance. The maintenance operations are classified according to the system restoration level after the operation. They can be divided into three classes.

3.1.2.1 Perfect maintenance

The first class refers to the so-called perfect maintenance. This kind of operations defines a complete replacement or a complete repair of the entity so that it goes back to an « As Good As New - AGAN » state. We then suppose that the entity goes back to its original state at the end of each operation. The operation can either be a corrective or a preventive maintenance operation. This hypothesis has been widely used in the literature [78]. It significantly eases the maintenance modelling and the performances evaluation.

3.1.2.2 Minimum maintenance

The second class refers to what we call minimum maintenance or minimum repair [20]. This kind of maintenance operation enables to only give back to the entity a part of its performances. We suppose that the entity recovers some properties to be back to the same functioning state as the one before the failure. The entity state after the operation is supposed to be « As Bad As Old - ABAO ». This repair type, generally associated with corrective maintenance, allows the entity to finish its mission while reducing at the maximum the repair time.

3.1.2.3 Imperfect maintenance

Finally, imperfect maintenance is the intermediate case between perfect and minimal maintenance. The imperfect maintenance operation aims at improving the entity health state by reducing its deterioration level but without bringing it back to an as good as new state. This kind of operation is suitable when an entity is used for a limited duration. Imperfect maintenance can then ensure the system availability on the remaining duration at a lower cost [125].

3.1.3 Maintenance model

A maintenance policy enables to decide which maintenance operations have to be done on the system and when they must be performed. This decision has to consider the costs generated by the maintenance operations, the production costs as well as the production loss costs. Indeed, the maintenance operations must be scheduled at the right time. If we do the replacements too early, we increase the global maintenance cost of the system with the multiplication of spare parts [54]. However, if the replacements are too late, we increase the risk of production loss with the system failure occurrences [185].

Scheduling a maintenance operation at the « best » moment becomes complex. Indeed, it depends on the current state of the system but also on the long-term consequences caused by this choice [150]. If the system evolution was known, it would be easy to identify the optimal maintenance policy for the system. However, it is impossible to know in advance the future behaviour of the system. It is then necessary to find another way to evaluate the maintenance decisions on the long term without knowing the future evolution of the system. To evaluate the maintenance policies performances and their consequences, we propose to use maintenance models.

Definition 12: *A maintenance model is a mathematical model that suggests a scenario for the system evolution and identifies the actions to do during this evolution.*

This model can report on the state evolution of a system subject to a maintenance policy and on quantifying the costs and gains generated by this policy. It is a decision support tool in maintenance and enables to choose the optimal maintenance strategy for the system.

Maintenance modelling enables to evaluate, validate or compare the performances of different maintenance policies that are not always easy to set up in practice.

3.1.4 Performance evaluation and optimization

To optimize the maintenance scheduling for an entity, it is essential to be able to evaluate the performances of the different policies that can be applied on it. This evaluation enables to define the optimal policy according to a chosen criterion. In the literature, we can generally

find three main criteria [78]: the economical criterion, the availability criterion and the safety criterion.

In high risk industries, like aeronautics or nuclear industries, a system malfunction can lead to very serious consequences. The maintenance policies that are applied to these systems are subject to severe norms and standards to avoid at all costs any failure occurrence. In this case, the safety is the selected criterion.

Nevertheless, the maintenance policy optimization is generally based on a cost criterion [32], [80], [98]. The objective is then to find the optimal distribution between corrective and preventive maintenance to maximize the positive effects of maintenance while minimizing the global cost. This cost includes the costs associated with the maintenance activities, such as inspections, replacements, logistics, and the costs due to the maintenance consequences, like over-maintenance, unavailability, towing, cargo loss. Figure 3.2 shows the effects of the maintenance operations frequency on the costs related to either corrective maintenance or preventive maintenance. Increasing the number of operations on the system enables to reduce the unwelcome effects caused by a failure but it penalizes the system functioning. It can also cause an increase of the system operating cost insofar as each maintenance operation generates a cost.

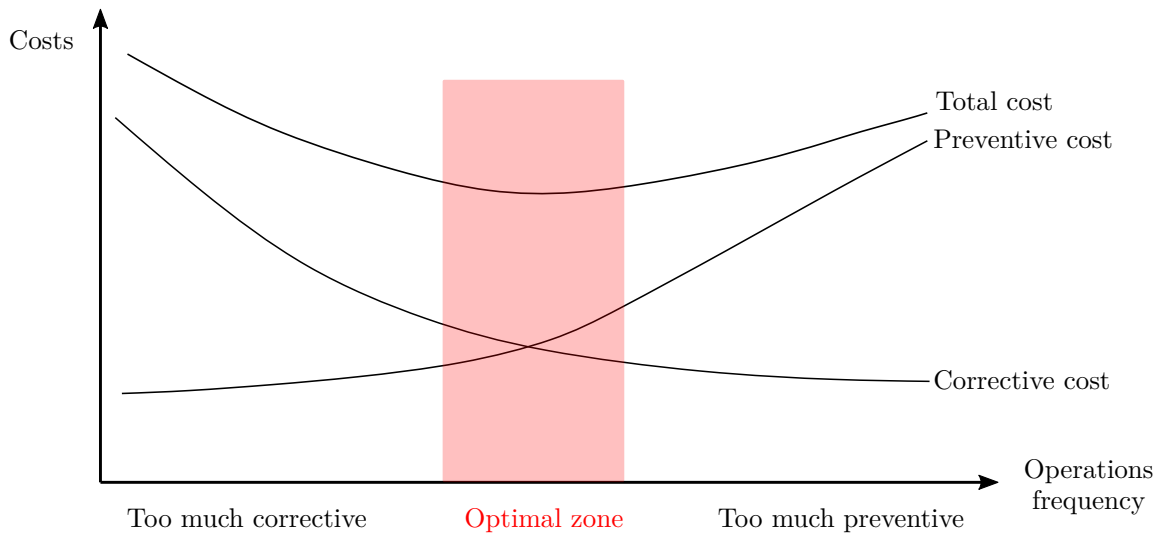


Figure 3.2: Balance between preventive and corrective maintenance

This definition of the global maintenance cost enables to integrate the availability issue in the decision-making process. Sometimes, it can be difficult to evaluate the costs related to the entity availability. In this case, a criterion only based on the availability can be considered [104].

In a general way, the cost criterion evaluation is strongly related to the considered time horizon, that can be finite or infinite. If the optimization horizon is supposed infinite, the average asymptotic cost is the most often used criterion to evaluate the maintenance policies performances [31]. This criterion is defined as the maintenance cost expectation by time unit.

If we denote $C(t)$ the global maintenance cost over a period t , the average asymptotic cost C_∞ is given by Eq. 3.1.

$$C_\infty = \lim_{t \rightarrow \infty} \frac{C(t)}{t} \quad (3.1)$$

If the replacements enable the entity to go back to its initial state, a renewing process can be considered [162]. The study of the average maintenance cost by time unit on an infinite horizon can then be reduced over a renewal cycle. This average asymptotic cost is equal to the ratio between the expectation of the cost on a renewal cycle and the average length of a cycle T_{cycle} (Eq. 3.2).

$$C_\infty = \lim_{t \rightarrow \infty} \frac{C(t)}{t} = \frac{\mathbb{E}[C(T_{cycle})]}{\mathbb{E}[T_{cycle}]} \quad (3.2)$$

In some cases, we can also use a finite horizon. The mathematical modelling is generally more complex to implement with a finite horizon. The challenge dwells in the length of the considered time interval because it affects a lot the relevance and the robustness of the results [97].

In practice, for easy maintenance policies, an analytical computation can be applied to evaluate the global maintenance cost. When the policies become more complex, numerical methods using stochastic simulation, especially the Monte Carlo method, are prioritized.

In our research work, we consider two economical criteria. The first one integrates the maintenance costs associated with corrective and preventive maintenance for a vehicle. The second one considers the gains generated by the vehicles missions completion and the delay costs associated with an overstepping of the missions deadlines.

3.2 Maintenance and monitoring systems

After reminding the general notions related to maintenance activities, we focus in this part on monitoring information and its links with maintenance. The monitoring operations aim at bringing information on the entity health state and on its usage conditions. They are mostly used for preventive maintenance policy such as condition-based maintenance and predictive maintenance policies. According to the quality of the available information, the operating state and the usage conditions can be either perfectly or partially known.

In a first stage, we will focus on the different existing types of monitoring information as well as the possible issues regarding their quality and the inspections scheduling. Then, we will explain the ways to integrate this information to make maintenance decisions.

3.2.1 Monitoring information

The monitoring domain is essential to build and optimize the decision-making process for a maintenance policy. It is reasonable to think that a maintenance decision is all the more relevant that it integrates a rich set of information regarding the system to maintain and its environment. As stated before, this information can be about the current health state of the monitored entity or about the environment in which the entity evolves. The growing use of monitoring in the maintenance domain has been possible thanks to the development of numerous technologies over the past few years. Indeed, they enable to have access to more and more accurate information on different systems. The challenge is then to find the good way to integrate them into the maintenance decision-making process.

3.2.1.1 Monitoring of the entity state

The monitoring of an entity depends on its complexity, on its environment, if it is easy to get access to it and if the right measurement tools are available. There are two types of monitoring: perfect and imperfect monitoring [78]. This clustering is based on the quality of the collected monitoring data.

In addition, beyond the quantitative aspect of the monitoring information, the availability is another issue to consider. The monitoring information can be available in a continuous or in a discrete way.

Perfect monitoring

Perfect monitoring enables to have access to an exact piece of information about the monitored entity. We then suppose that the collected data provide a reliable and accurate piece of information. In practice, we cannot check that we have perfect monitoring. However, it is not about considering an ideal hypothesis but rather that measurement errors are small enough so that their impact on the data can be neglected. Perfect monitoring is often the result of a direct monitoring [40]. However, monitoring can still be perfect even if we do not have a direct monitoring. The measurements must be accurate enough so that they can be treated and synthesized to obtain a health state indicator for the monitored entity. An example of perfect indirect monitoring is realized on the engine oil of commercial heavy vehicles [80]. The hypothesis of perfect monitoring has been widely used in the literature [51], [111], [177].

Imperfect monitoring

However, it happens that perfect monitoring does not correspond to the reality. In this context, the issues related to imperfect monitoring have to be considered [21]. Sometimes, the complexity of the entity, the monitoring measurements inaccuracy or its environment can lead to information tainted with errors on the entity current status [34], [183]. Most of the time,

we are facing classical diagnosis mistakes such as false alarm, detection delay, non-detection or measurement errors. The developed maintenance policy will then have to consider these errors in the maintenance decision-making process. For instance, if a technical test indicates that the entity reached a fixed threshold, it can be true. But there is a probability that it is a false alarm. The decision process should then consider this probability. Here, the challenge is to know how to use this imperfect information to make a better decision than if we did not have it.

Continuous monitoring

The next issue regarding monitoring data is to know how often they are available. We can either have continuous monitoring or discrete monitoring [70]. For continuous monitoring, we consider that the entity current state is available at any time. This kind of monitoring strategy is set up for critical entities or when the monitoring costs are relatively low. Continuous monitoring is considered in the research studies on maintenance problems with constraints on the possible times for interventions or delays between an alarm occurrence and an intervention [33], [81].

Discrete monitoring

On the contrary, if monitoring information is only available at specific moments, through inspections, the maintenance issues are quite different. Each inspection can range from a basic visit to a more sophisticated control operation. The inspection type choice can help to optimize the maintenance policy. During an inspection, two decisions have to be made. The first one is to decide if a maintenance operation is necessary, given the entity current state. Indeed, most of the time, the maintenance operations, especially the preventive ones, coincide with inspections. The second decision to make is to set the next inspection date.

It is then necessary to schedule at best the inspections as we leverage on them to optimize the implemented maintenance policy. Different inspection schemes are considered in the literature. The first one is a periodic inspection scheduling that is independent of the health state evolution of the monitored entity [40]. The objective is then to determine the optimal inspection period according to the chosen maintenance policy or, on the contrary, to define the optimal policy according to the inspection period [131]. There are also more complex schemes in which the inspections are dynamically scheduled according to the entity state evolution [71]. They are especially suitable for a deteriorating entity. Note that if an inspection does not affect the entity state, it is a conservative inspection. On the contrary, if it alters the entity state, it is considered as a destructive inspection.

3.2.1.2 Monitoring of the environment

The previous classification refers to monitoring information related to the entity current state (operating state, breakdown, deterioration level). However, monitoring systems also enable

to obtain information related to the operating environment in which the entity evolves. This information is essential for systems evolving in a dynamic environment, like industrial vehicles. Indeed, the significance of the entity current state is different whether the entity evolves in a stressed environment or not. Monitoring a set of parameters correlated with the entity deterioration evolution become essential to properly interpret the entity health measurements. Typically, for an industrial vehicle, we consider the topography, the road surface condition, the temperature, the load, and so on.

3.2.1.3 Monitoring cost

Having access to monitoring information can be quite costly according to the accuracy we want regarding the entity health state and/or the operating environment. It is necessary to consider this cost in the evaluation and in the optimization of the maintenance policy. It is all about knowing if the investments granted in the monitoring architecture are compensated by the benefits obtained in terms of maintenance.

In our work, when we study the static maintenance and missions scheduling case, we consider that we do not have any monitoring information regarding the vehicle health state. On the contrary, with the dynamic case, we consider that deterioration information can be available but not at any time. We are then not in the case of a continuous monitoring. However, we consider that the monitoring is perfect and there are no errors in the collected deterioration data. Based on this information and by comparing the dynamic and the static cases, we can evaluate the gains earned when using health monitoring systems. It will be the subject of a part in a following chapter.

3.2.2 Impact of the monitoring information on the decision-making process

To implement and optimize a condition-based or a predictive maintenance policy on a specific entity, monitoring information plays a key role. It enables to provide to the decision makers knowledge to ease the decision-making process. This part aims at showing the way we can consider the monitoring information to build an efficient decision-support tool.

For condition-based maintenance, maintenance operations are triggered when parameters related to the entity deterioration appear. Most of the time, the maintainer relies on the entity deterioration level. This indicator, widely used in the literature [63], [175], has the advantage of being directly usable without any other transformation. However, it does not take into account the environment in which the entity evolves. For an entity operating in a dynamic and very changing environment, this information can have a significant impact on the interpretation of the deterioration measurement.

Actually, the deterioration level helps to understand the past and the present of a considered entity to make a maintenance decision. However, the maintainer needs to schedule the operations to come. It is then useful to complete the information based on the entity

past and present with a characterization of its future evolution. This statement justifies the development of predictive maintenance strategies that integrate the entity future deterioration evolution to make decisions. Most of the time, the predictive approach integrates the hypotheses made on the future usage of the entity. This type of preventive maintenance has become quite famous for the past few years. Many studies emphasize its interest and show how beneficial this type of maintenance is to optimize the decision making process [31], [80], [97]. It then seems interesting and valuable to develop maintenance policies that can consider this predictive aspect.

Now that we drew a distinction between condition-based maintenance and predictive maintenance, we can illustrate the impact of the monitoring information on the decision-making process. To do so, we will estimate the remaining useful life (RUL) of an entity according to the available monitoring information. The objective is to show how monitoring data can influence the maintenance decision.

Let us consider that an entity deterioration follows a Gamma process whose parameters are $\alpha = 1$ and $\beta = 1$ with a failure threshold fixed at $L = 100$. Estimating the remaining useful life of an entity corresponds to estimate its reliability conditionally to its observed deterioration level. Knowing that at time t_m , the entity deterioration level is $Z(t_m)$, the conditional reliability can be defined as in Eq. 3.3.

$$R(t|Z(t_m)) = 1 - \frac{\Gamma(\alpha(t - t_m), (L - Z(t_m))\beta)}{\Gamma(\alpha(t - t_m))} \quad (3.3)$$

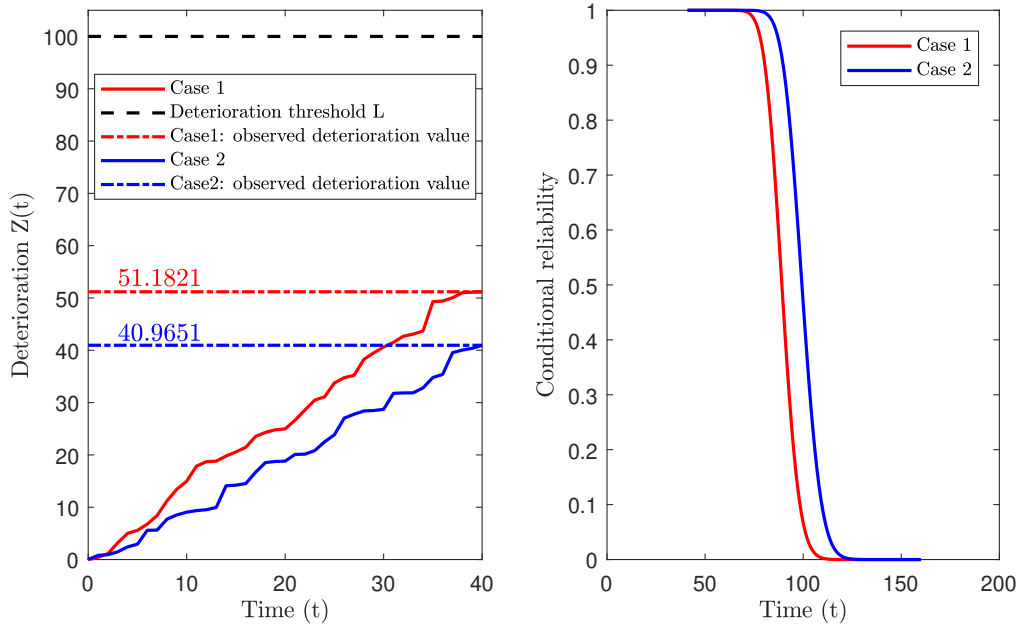


Figure 3.3: Update of the conditional reliability based on monitoring information

Figure 3.3 presents the conditional reliability estimation for two different deterioration trajectories. In both cases, the deterioration level knowledge enables to influence the remaining useful life estimation. Indeed, the RUL can be updated according to the real state of the entity. It is more likely that the failure occurs in case 1 before occurring in case 2. It shows that the maintenance decision can possibly be modified thanks to monitoring information.

3.3 Maintenance policies for mono-component systems

We have previously defined the framework when working with maintenance policies and the ways to optimize them. In this section, we will detail the main preventive maintenance policies used to maintain a single-component system. These policies can be clustered into two groups according to the kind of information used to make maintenance decisions [31], [66], [97]. The first class groups the policies based only on the a priori properties of the component. We then use the available component lifetime model or the component deterioration model to think in terms of failure occurrence probability. This probability corresponds to the probability to have a failure at each time or to be in a given deterioration state over time. The second class gathers all the policies based on the component deterioration level. In this case, the maintenance decisions are always supported by failure models but are completed by monitoring information on the component current state. The two classes present two maintenance aspects: the static one with the first cluster and the dynamic one with the second cluster. This state of the art is built upon many research studies from the literature [154], [166], [174].

3.3.1 Maintenance policies based on the system lifetime law

Barlow and Proschan [20] identify three kinds of maintenance policies based on the knowledge of the system properties:

- Age replacement policies,
- Periodic replacement or block replacement policies,
- Inspection-based policies (inspections enable to detect the system failure but no information is available regarding the system state).

For this policies cluster, we suppose that the system lifetime law for normal operating conditions is known or at least partially known. It enables to schedule the replacement moments. The maintenance actions are then planned at fixed deadlines without considering the system wear state. We then have static interventions. Optimizing these policies is about identifying the optimal replacement date to minimize the defined cost criterion.

3.3.1.1 Age replacement policy

For age replacement policies, the guideline is to avoid at best the component failure by suggesting to replace it when its age reaches T_0 time units [137]. The age is actually the time elapsed since the last replacement. To optimize this strategy, we have to find the time interval T_0 offering the best compromise to minimize the maintenance cost per time unit on an infinite horizon. T_0 cannot be too small because it would lead to an increase of the maintenance cost due to too many preventive maintenance operations. However, it cannot be too high either to avoid increasing the number of failures.

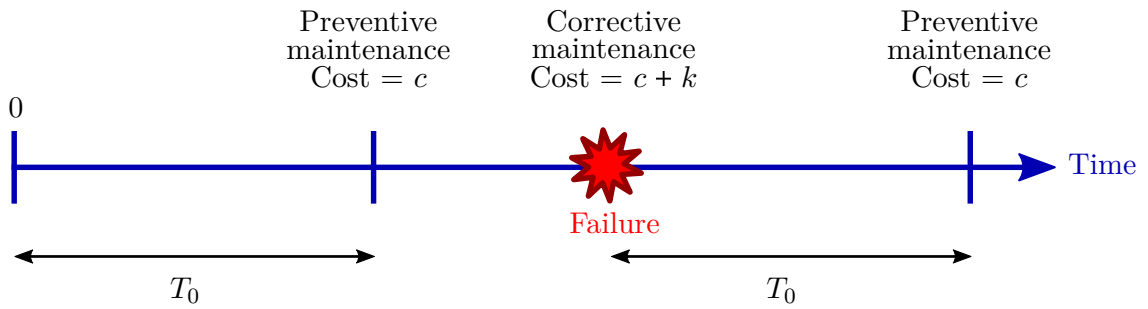


Figure 3.4: Age replacement policy and costs

Age replacement consists in replacing the component (Figure 3.4):

- when it reaches the operating age T_0 with a unitary preventive replacement cost equal to c ,
- When there is a failure before T_0 with a unitary corrective replacement cost equal to $c + k$ (the failure generates an additional cost k).

Based on this definition, we can compute the average maintenance cost per time unit on a infinite horizon $\overline{C_A}$. This asymptotic cost is defined in Eq. 3.4.

$$\overline{C_A} = \lim_{t \rightarrow \infty} \frac{\mathbb{E}[C_A(t)]}{t} = \lim_{t \rightarrow \infty} \frac{(c + k)\mathbb{E}[N_c(t)] + c\mathbb{E}[N_p(t)]}{t} \quad (3.4)$$

where $\mathbb{E}[C_A(t)]$ is the expectation of the maintenance cost at time t , $N_c(t)$ and $N_p(t)$ are respectively the number of corrective and preventive replacements occurring on the time interval $[0; t]$. If we consider that these replacements bring back the system to its initial operation condition, a renewal process is generated with this policy. It is then possible to use the renewal theorem. In this case, we consider a renewal cycle to compute the total average maintenance cost per time unit on an infinite horizon.

Let T be the component failure time and T_0 the time when the replacement is scheduled. On a renewal cycle, the expectation of the maintenance cost is defined by Eq. 3.7.

$$\mathbb{E}[C_A(T_{cycle})] = (c + k)P(T \leq T_0) + c.P(T > T_0) \quad (3.5)$$

$$= (c + k)F(T_0) + c.R(T_0) \quad (3.6)$$

$$= c + k.F(T_0) \quad (3.7)$$

with F and R respectively the failure distribution function and the survival function of the component.

The average length of the renewal cycle verifies Eq. 3.10.

$$\mathbb{E}[T_{cycle}] = \int_0^{T_0} t.f(t)dt + T_0P(T \geq T_0) \quad (3.8)$$

$$= \int_0^{T_0} (1 - F(t))dt \quad (3.9)$$

$$= \int_0^{T_0} R(t)dt \quad (3.10)$$

The average maintenance cost per time unit on a infinite horizon is then equal to:

$$\overline{C_A} = \frac{\mathbb{E}[C_A(T_{cycle})]}{\mathbb{E}[T_{cycle}]} = \frac{c + k.F(T_0)}{\int_0^{T_0} R(t)dt} \quad (3.11)$$

The optimal replacement time interval T_0^* is obtained by minimizing the average maintenance cost per time unit on a infinite horizon such that:

$$\frac{d\overline{C_A}}{dT_0}(T_0^*) = 0 \quad (3.12)$$

We denote the average optimal asymptotic cost of this policy C_A^* , reached with the optimal replacement time interval T_0^* (Eq. 3.13).

$$C_A^* = \overline{C_A}(T_0^*) \quad (3.13)$$

We can illustrate the previous results with an example. Let us consider a component whose lifetime is modelled with a Weibull law. The shape and scale parameters are respectively $\alpha = 5$ and $\lambda = 3000$. We suppose that the preventive cost is $c = 500\text{€}$ and the additional cost associated with corrective maintenance is $k = \{700\text{€}, 2000\text{€}, 5000\text{€}\}$. Figure 3.5 presents the evolution of the average maintenance cost for different values of additional cost k . The higher the value of k is, the smaller the optimal replacement interval T_0^* . It is due to the fact that increasing the value of k leads to increase the unitary corrective maintenance cost. It is then wiser to avoid performing corrective maintenance operations. By fixing a smaller replacement interval, we avoid at best the failures.

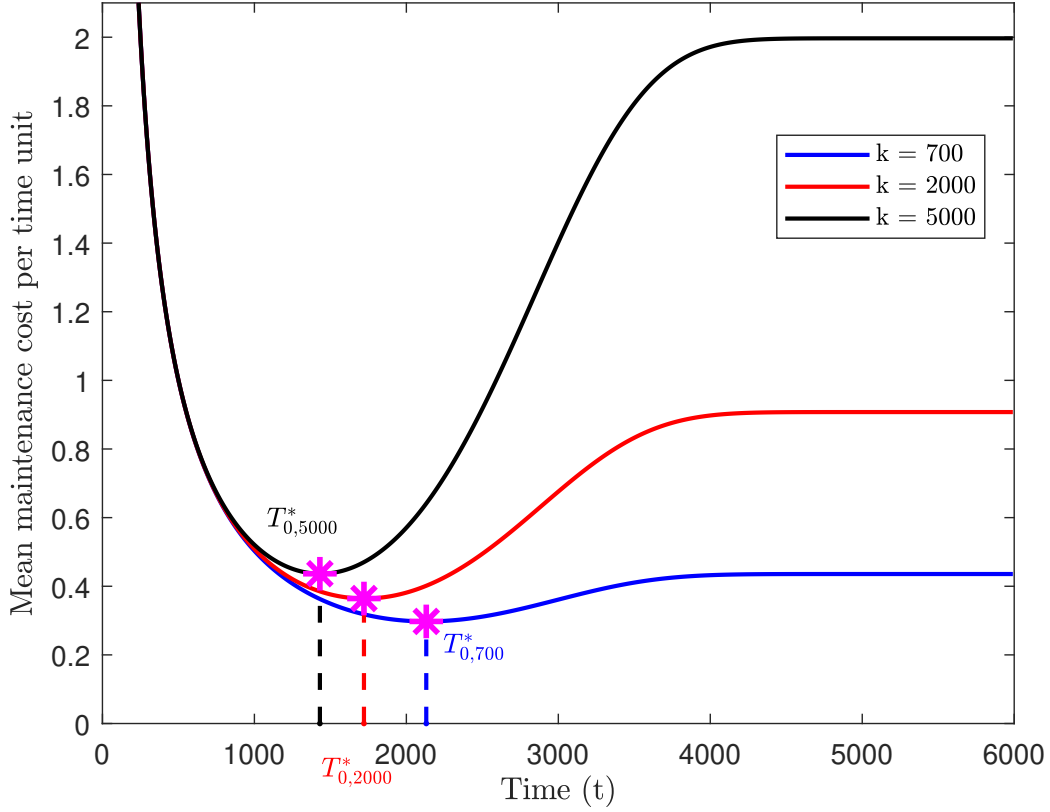


Figure 3.5: Mean asymptotic maintenance costs for an age replacement policy

To apply an age replacement policy, we need to monitor the system age. In practice, it can be difficult because we can face struggles when it comes to collect, process and store information, in particular on complex systems. Moreover, as the maintenance operations are scattered over time, it is impossible to define a long-term schedule for the operations. Indeed, each failure on the system leads to postpone all the following operations. However, Barlow and Proschan showed that this policy is the best one in the random periodic policies class [20].

If considering the perfect maintenance hypothesis is not realistic, other authors have worked on age replacement policies by integrating minimum or imperfect repairs [120], [133], [166] and variable maintenance operations costs [151].

3.3.1.2 Block replacement policy

The block replacement policy consists in replacing the component by a new one at regular intervals whose period is T_0 without consideration for the failure history [18]. It is different from the age replacement policy insofar as the decision rule is not rebooted after a failure.

It means that the preventive maintenance program at $T_0, 2T_0, 3T_0$, and so on, remains unchanged despite failure occurrences (Figure 3.6). The component is also replaced when there is a failure before the next scheduled maintenance intervention. This strategy can become quite expensive as we replace the component both at the preventive interventions at time $i.T_0$, with $i \in \mathbb{N}$, and at the failure occurrences.

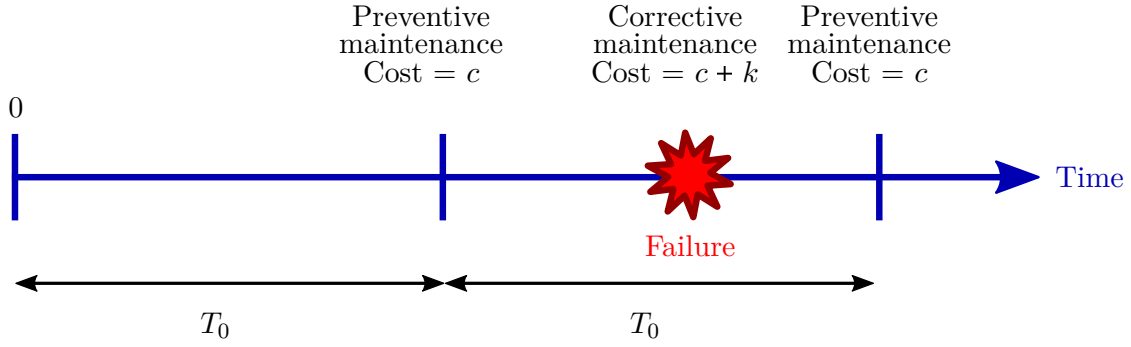


Figure 3.6: Block replacement policy and costs

The optimal replacement time interval T_0^* is obtained by minimizing the cost criterion. As in Section 3.3.1.1, we consider $\overline{C_B}$ the average maintenance cost per time unit on an infinite horizon, as described in Eq. 3.14, where $\mathbb{E}[C_B(t)]$ is the maintenance costs expectation value at time t for a component subject to a block replacement policy.

$$\overline{C_B} = \lim_{t \rightarrow \infty} \frac{\mathbb{E}[C_B(t)]}{t} \quad (3.14)$$

As we consider that we replace the component by a new one, we can use the renewal theorem and reduce the study on a renewal cycle, whose length is equal to T_0 . On this T_0 period, the expectation of the maintenance cost is defined as in Eq. 3.15.

$$\mathbb{E}[C_B(T_0)] = c + (c + k) \cdot \mathbb{E}[N_c(T_0)] \quad (3.15)$$

$\mathbb{E}[N_c(T_0)]$ is the expected value of the failure occurrences on the period T_0 . If we assume that the component only fails once over the time period equal to T_0 , we can approximate the number of failures $\mathbb{E}[N_c(T_0)]$ to the failure probability $F(T_0)$. We then obtain the average asymptotic maintenance cost such that:

$$\overline{C_B} = \frac{\mathbb{E}[C_B(T_0)]}{\mathbb{E}[T_0]} \quad (3.16)$$

$$\approx \frac{c + (c + k) \cdot F(T_0)}{T_0} \quad (3.17)$$

The optimal replacement period T_0^* is identified by minimizing this average asymptotic cost (Eq. 3.18).

$$\frac{d\overline{C_B}}{dT_0}(T_0^*) = 0 \quad (3.18)$$

We denote C_B^* the average asymptotic optimal cost for this policy. It is reached for the optimal replacement period T_0^* (Eq. 3.19).

$$C_B^* = \overline{C_B}(T_0^*) \quad (3.19)$$

The block replacement policy is easier to manage than an age replacement policy. It is possible to define a long-term schedule for the maintenance operations and this schedule is stable over time. However, this policy is not the most cost-effective as it can lead to the replacement of almost new components. To reduce the consequences generated by this risk, it is possible to consider intermediate repair levels.

Berg and Epstein [25] suggest to change the system for a new one when it fails. However, if at the scheduled replacement time $i.T_0, i \in \mathbb{N}$, the system age is less or equal to $T_0^* + \omega$, with $-T_0 < \omega < 0$, then the system is not replaced. It goes on operating until it fails or until the next scheduled replacement at $(i + 1).T_0$. They show that this method enables to reduce the average maintenance cost per time unit on a infinite horizon with respect to the initial strategy.

Tango [159] proposes to replace the component by a new one at the periodic intervals $i.T_0, i \in \mathbb{N}$. If the component fails before the scheduled replacement date, it is replaced by a component that is not new. These « non-new » components are components that have been preventively replaced and were still operating during the replacement. The advantage when using « non-new » components is to consider that the spare part cost is equal to 0 when estimating the maintenance cost. The average asymptotic maintenance cost is then expressed per time unit to find the optimal period T_0^* to minimize the criterion. These two aspects have to be considered to optimize the inspection-based policy.

3.3.1.3 Inspection-based policy

There are systems whose failures can only be identified through inspections. These systems cannot be continuously monitored. An inspection model assumes that the component state is completely unknown except if an inspection is carried out. Each inspection is supposed perfect and retrieves the system state without error. After each inspection, two decisions must be made. The first one is about the maintenance operation to perform. Should we replace the component or let it in its current state? The second decision is about scheduling the next inspection.

Barlow et al. [19] developed a model only based on inspections. It means that no preventive replacement is performed. The component is only replaced when it fails. The model assumes that the failure is identified during the inspections. These inspections do not deteriorate the component and the component cannot fail during one of them. The inspections

are carried out at times $T_{j \in \{1, \dots, n\}}$. We assume that a failure occurs at time T such as $T_{n-1} < T \leq T_n$ (Figure 3.7).

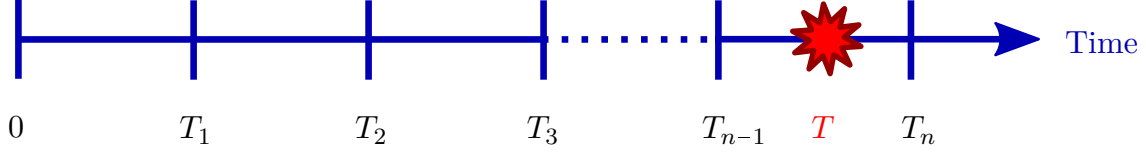


Figure 3.7: Inspection-based policy

As the replacements are new, the average maintenance cost on an infinite horizon \overline{C}_I is equal to the expected value of the maintenance cost on the average length of the cycle inspection $[0; T_n]$. The maintenance cost expectation on an inspection cycle is defined in Eq. 3.20.

$$\mathbb{E}[C_I(T_n, T)] = c_i \cdot n + c_d(T_n - T) \quad (3.20)$$

where c_i and c_d are respectively the unitary inspection cost and the cost by time unit when the system is in a degraded mode with an undetected failure. To obtain the optimal inspections dates T_j^* , we have to minimize \overline{C}_I such that:

$$\overline{C}_I = \frac{\mathbb{E}[C_I(T_n, T)]}{\mathbb{E}[T_n]} \quad (3.21)$$

Other maintenance policies have been developed based on this model [166].

3.3.2 Policies based on the current system state

After presenting the different existing policies based on the component lifetime, we will focus on the maintenance policies based on the current component state. For the component that fails after a gradual deterioration, the component state evolves from a new state through different intermediate states before failing. Characterizing the component only with its lifetime law is no longer sufficient. We suppose in this part that we have access to the component deterioration. It means that monitoring information related to the deterioration is available.

Based on this information, the maintenance decisions can be adapted. It then allows the maintenance policies presented in Section 3.3.1 to evolve towards more dynamic policies. In this context, the condition-based maintenance policy and the predictive maintenance policy can be introduced.

3.3.2.1 Condition-based maintenance

The condition-based maintenance policy aims at better controlling the component behaviour by monitoring its deterioration. This objective can be seen as a way to increase the operating time of the component, to reduce the maintenance costs and to improve the safety [30]. This policy is mostly used on components considered as critical ones in terms of safety and costs.

According to Boulenger [30], condition-based maintenance brings many advantages such as:

- an increase of the component lifespan,
- a better managed control of the component,
- a less expensive repair cost,
- a spare parts limitation,
- a safety improvement,
- an improvement in the credibility of the maintenance services.

However, applying condition-based maintenance on a component requires the use of an equipment to measure the level of one or more deterioration indicators. The component deterioration must then be quantifiable, either with physical or software sensors, or through component inspections made by a technician.

The usual scheme of condition-based maintenance policy is depicted in Figure 3.8. As explained before, the maintainer makes his decision based on the observed deterioration level [40]. It is then necessary to schedule regular inspections at the right time to check the current component state, denoted Z . On Figure 3.8, we consider that the inspections are performed at a regular time interval τ . For condition-based maintenance, each inspection is seen as an opportunity to perform a maintenance operation. Indeed, at each inspection, a maintenance operation is carried out if the deterioration level Z exceeds the preventive threshold Z_{prev} . A preventive replacement costs c and we also add the cost c_{insp} associated to the inspection. On the contrary, if $Z < Z_{prev}$, the component is normally operating and we just consider the inspection cost c_{insp} . Finally, the component is considered as failed when Z exceeds the limit threshold L , fixed by the system supplier. In this case, a corrective cost $c + k$ is charged.

Based on the previous description, the maintenance cost at each time t can be defined (Eq. 3.22).

$$C_c(t) = N_{insp}(t).c_{insp} + N_c(t).(c + k) + N_p(t).c \quad (3.22)$$

with $N_{insp}(t)$, $N_c(t)$ and $N_p(t)$ respectively the number of inspections, the number of corrective and preventive replacements at time t . Thanks to the renewal theory, the average

maintenance cost per time unit on an infinite horizon \overline{C}_c can be computed in the same way as with the previous maintenance policies, based on the cycle T_{cycle} (Eq. 3.24).

$$\overline{C}_c = \lim_{t \rightarrow \infty} \frac{\mathbb{E}[C_c(t)]}{t} \quad (3.23)$$

$$= \frac{\mathbb{E}[C_c(T_{cycle})]}{\mathbb{E}[T_{cycle}]} \quad (3.24)$$

For a given deterioration process, the values N_{insp} , N_c and N_p as well as the cycle T_{cycle} depend on the decision variables. These variables are the preventive threshold Z_{prev} and the inspection period τ . The average asymptotic maintenance cost depends not only on these decision variables but also on the inspection cost c_{insp} . To obtain the optimal asymptotic cost C_c^* , it is necessary to find the best tuning for these decision variables by using a maintenance model.

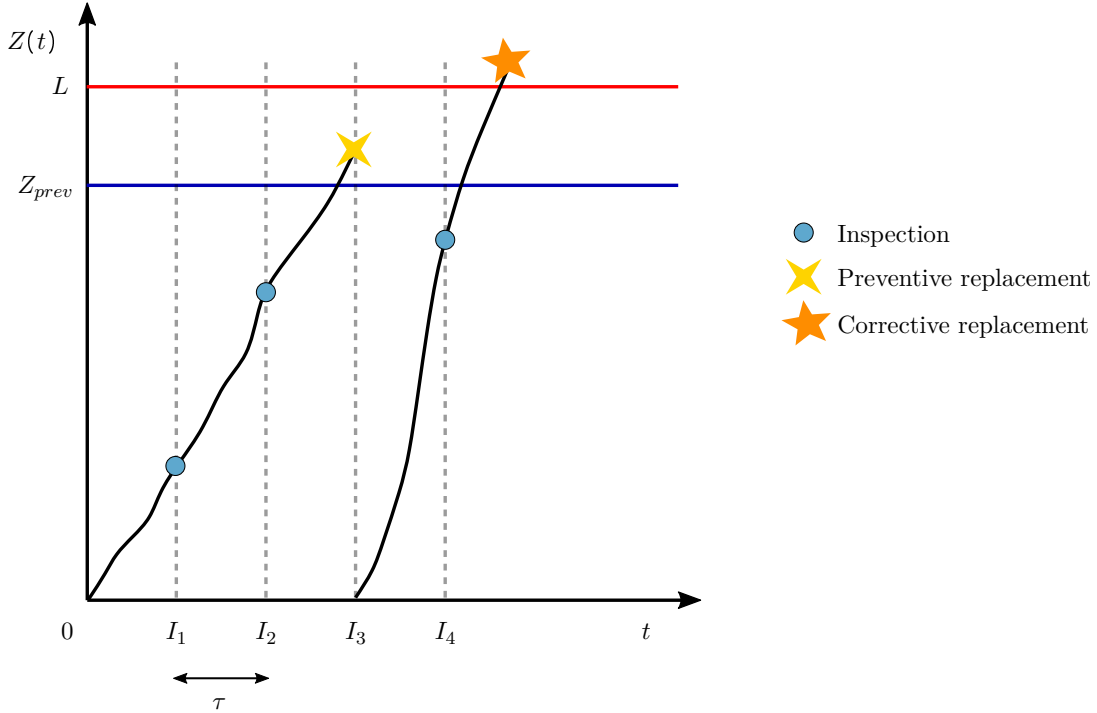


Figure 3.8: Condition-based maintenance policy

3.3.2.2 Predictive maintenance policy based on the remaining useful life

The predictive maintenance approach enables to consider a prediction on the health indicators or on a system feature in the decision-making process.

The remaining useful life can integrate different information such as the deterioration level or the future state of the environment, if it is known. In this part, we describe a predictive

maintenance policy that consists in integrating the component remaining useful life in the decision-making process. This risk-based policy [80] aims at using prognosis in a direct way to make maintenance decisions.

This policy aims at ensuring that the component failure risk never exceeds a fixed risk threshold r . It means that, at every time, the component conditional reliability must overstep the value $1 - r$. This risk threshold has to be optimized using the maintenance model or has to be given by the operational constraints.

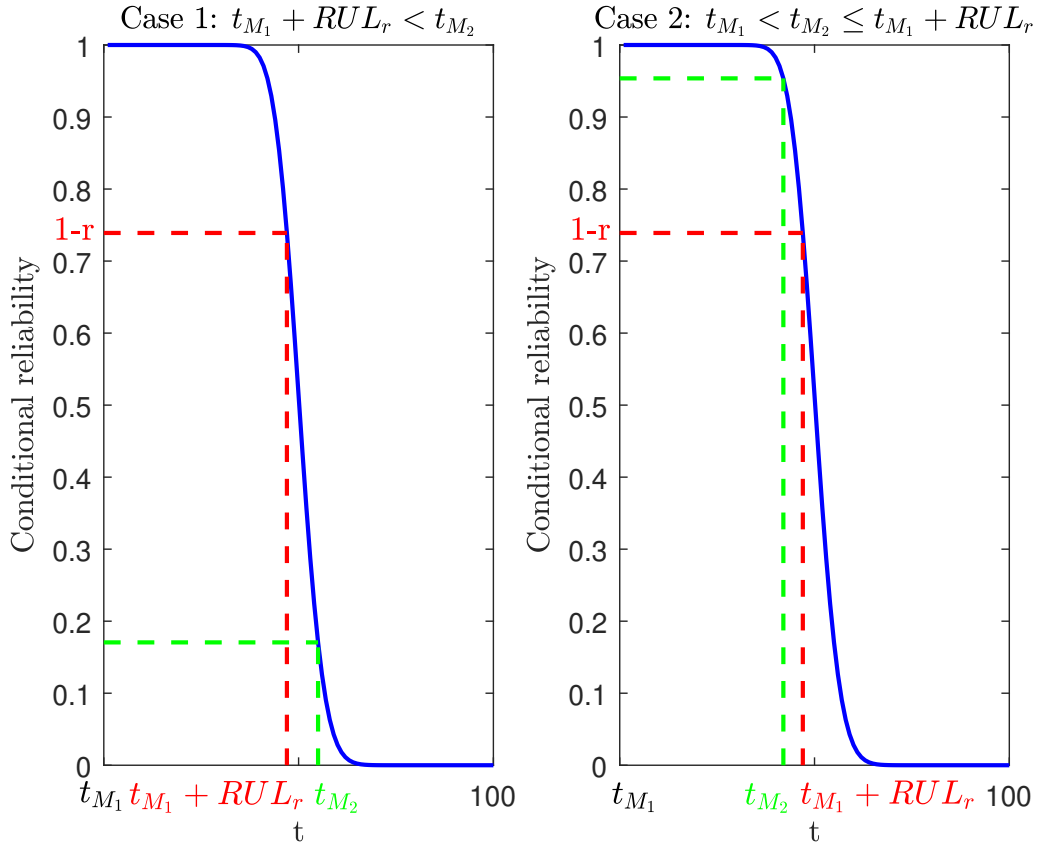


Figure 3.9: Predictive maintenance policy based on the risk

Let us consider that we know the next two maintenance opportunities for a component. They will occur at t_{M_1} and at t_{M_2} . If the remaining useful life estimation done at t_{M_1} indicates that the risk constraint is not respected for the second maintenance opportunity occurring at t_{M_2} , then a preventive replacement is performed at t_{M_1} . Otherwise, the preventive replacement is not scheduled. In practice, to make a maintenance decision, it is about adding the component remaining useful life for a given risk RUL_r with t_{M_1} , and to compare it with the date t_{M_2} (Figure 3.9). If $t_{M_1} + RUL_r < t_{M_2}$, the component remaining useful life is not enough to reach t_{M_2} . A preventive replacement is then scheduled at t_{M_1} . Otherwise, the failure risk remains inferior to r until the next maintenance opportunity at t_{M_2} .

It is also possible to consider a predictive maintenance policy not based on the risk but

on the costs. In this case, the choice regarding the maintenance opportunity is made for the one generating the lowest cost.

The dynamic maintenance policies presented in [Section 3.3.2](#) are more efficient than the age replacement policy or the block replacement policy because the decisions regarding the maintenance actions are based on the real component deterioration. However, condition-based maintenance and predictive maintenance involve an additional cost to have access to monitoring information. It is then necessary to evaluate the dynamic policies profitability by using a maintenance model.

3.4 Maintenance policies for multi-component systems

In this part, we will present the major existing maintenance policies for multi-component systems. Regarding the decision-making process, the problem is quite different between mono-component systems and multi-component systems. Indeed, optimal maintenance decisions for components considered separately are not necessarily the best ones when it comes to consider them together to fulfil a common function [\[21\]](#). Of course, if all the components are completely independent from one another, we can just applied individually the policies described in [Section 3.3.1](#) on each component. However, most of the time, optimal maintenance decisions for a multi-component system cannot be just a simple concatenation of the decisions made for each component independently.

In a multi-component system, the maintenance decision is supported by the proper operating mode of each component (lifetime model, deterioration model, ...), the system structure (series, parallel, ...) and the existing dependencies between the components. These dependencies can be economic, structural or stochastic [\[161\]](#).

We have economic dependencies when the cost of a maintenance operation for a group of components is different from the total maintenance cost generated if the operations are performed on each component individually. This dependence effect appears in the maintenance operations performing date. Grouping maintenance operations aims at saving logistic costs related to the system entrance in the maintenance phase. When it is impossible to maintain a component without entailing consequences on other components, we talk about structural dependencies. For instance, if a component fails, it is necessary to dismantle other components to be able to maintain the failed one. This dependency is a constraint when it comes to schedule the maintenance operations [\[91\]](#) but it can also be seen as an opportunity to replace other components already dismantled [\[39\]](#). We talk about stochastic dependence when a component failure has an influence on the lifetime law or on the deterioration phenomenon for other components [\[145\]](#). The failure of a component can affect the failure on another one and so on.

In the following sections, we describe the major grouping methods used to minimize the maintenance cost of multi-component systems. By relying on the state of the art available regarding the maintenance policies for multi-component systems [\[10\]](#), [\[31\]](#), [\[39\]](#), [\[47\]](#), [\[97\]](#),

[123], [124], [181], it is possible to differentiate two grouping categories: the static grouping strategies and the dynamic grouping strategies.

3.4.1 Static grouping policies

The static maintenance grouping policies schedule the maintenance operations on a long-term horizon by minimizing a given criterion that is, most of the time, based on the maintenance cost. They are referred to as static policies insofar as the maintenance decisions are made on the assumption that the operating environment remains stable over an infinite horizon. Indeed, they do not use the information on the current system state to update the initial schedule. It means that the maintenance rules will not change during the system lifetime.

The static grouping strategies are divided into three categories:

- the corrective maintenance grouping strategy,
- the preventive maintenance grouping strategy,
- the opportunistic maintenance strategy.

3.4.1.1 Corrective maintenance grouping

The corrective maintenance grouping strategy consists in waiting the initially scheduled maintenance operation to replace the failed components. The maintenance cost is reduced by grouping these maintenance operations at the same date because it enables to save logistic costs related to the system entry in the maintenance phase. By reducing the number of maintenance interventions, we can save a part of the costs generated by the maintenance activity. However, the costs related to the unavailability of the failed components need to be compensated. This type of policy is aimed at systems with redundant functions such as a production plant with machines operating in parallel. The objective is then to find a balance between the gain generated by the logistic costs savings and the production loss due to the failed machines. Many authors [14], [79], [127] have worked to identify the limit number of components or machines for which this policy remains interesting.

3.4.1.2 Grouping of scheduled preventive maintenance operations

Maintenance policies to group preventive maintenance operations are generally based on the maintenance policies developed for the elementary components. This includes age-based policies, block replacement policies and inspection-based policies.

Age-based maintenance policy

This policy aims at replacing groups of components at defined time intervals. A component group is also maintained when one of the component in the group fails. Contrary to a block replacement policy, we do not fix periodic time intervals for the maintenance operations at time $T_0, 2T_0, 3T_0$. Each component can then have any replacement frequency, determined according to the age-based maintenance policy. Firstly, the frequency of the calendar replacements is optimized for each component. Then, the objective is to modify the frequency to group several maintenance operations at the same time. Identifying the maintenance operations groups and their associated maintenance intervals enable to obtain a optimal policy for the whole system. This scheduling aims at optimizing the average maintenance cost per time unit on an infinite horizon by taking advantage of the economic dependencies to save logistic costs.

Van Dijkhuizen and Van Harten [167] assume that every component i of the system has an optimal known maintenance age x_i^* . They consider that the optimal maintenance age $x_{G_j}^*$ for an operation group G_j is defined as $x_{G_j}^* = \min_{i \in G_j} (x_i^*)$. Then, they use a dynamic algorithm to identify the partition of component groups for which the average maintenance cost per time unit over an infinite horizon is minimum. This problem becomes more complex when different logistic costs are introduced. In this case, a heuristic method is applied.

Block replacement policy

The block replacement strategy for a multi-component system consists in replacing component groups at regular time intervals $T_0, 2T_0, 3T_0$ without considering the failure history for the components in the group. These component groups are maintained at scheduled dates and when a failure on one of the components in the group occurs. To optimize this policy, we have to identify the groups of components to minimize the average maintenance cost per time unit on an infinite horizon. As with a single component, work is being done to adapt this policy. The solutions aim especially at introducing minimum repairs but also an age constraint to manage the replacement of failed components.

Archibald and Dekker [12] develop an extension of the method from Berg and Epstein [25] for a multi-component system. They assume that, if a component i in the group G_j fails before the date $k.T_0$ with $k \in \mathbb{N}$, the component i is replaced at the failure date if and only if its age is less than $T_0 + \omega_i$, with $-T_0 < \omega_i < 0$. Otherwise, the component i is not replaced and the system operates in a degraded mode. The objective here is to find the replacement period T_0 for the groups G_j of maintenance operations and the ages $T_0 + \omega_i$ for the components i that minimize the average maintenance cost per time unit for the whole system on an infinite horizon.

Ait Kadi and Cl  roux [6] depict a block replacement policy for which the failing components can be replaced by « non-new » components. This policy (Figure 3.10) is based on the following rules:

- the group G_j of components is preventively replaced with « new » components at time $k.T_0, k \in \mathbb{N}$,

- if the component i fails in the time interval $[(k-1).T_0; k.T_0 - \omega_1)$, $k \in \mathbb{N}$, the component is replaced by a « new » one,
- if the component i fails in the time interval $[k.T_0 - \omega_1; k.T_0 - \omega_2)$, $k \in \mathbb{N}$, $0 \leq \omega_1 \leq \omega_2 \leq T_0$, the component is replaced by a « non-new » one,
- if the component i fails in the time interval $[k.T_0 - \omega_2; k.T_0)$, $k \in \mathbb{N}$, the component remains inactive until the scheduled date $k.T_0$, $k \in \mathbb{N}$

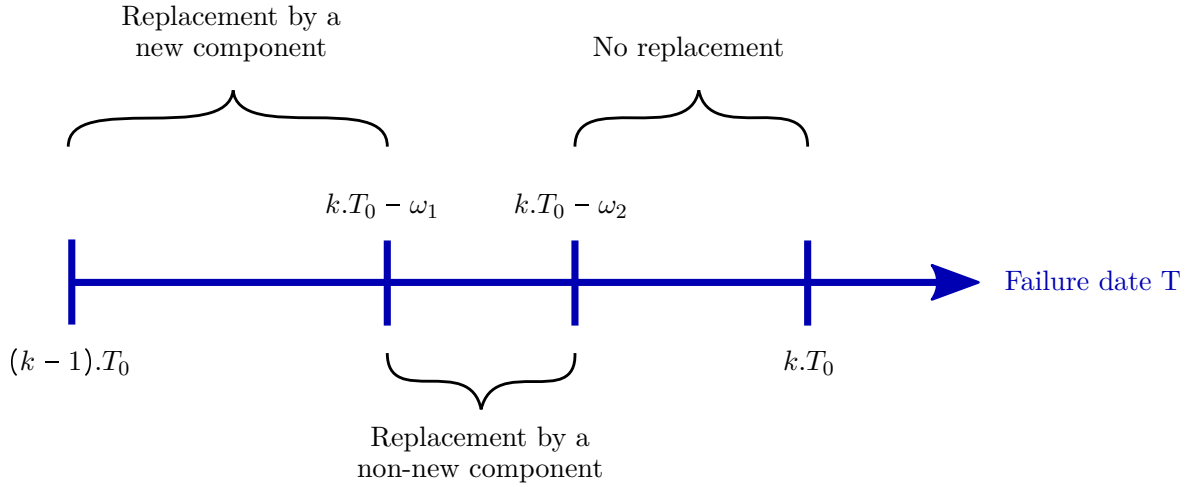


Figure 3.10: Grouped block replacement policy [6]

Then, they develop an algorithm to identify the best values for T_0, ω_1, ω_2 to minimize the average maintenance cost per time unit on an infinite horizon.

Inspection-based policy

The inspection-based policies for multi-component systems consist in inspecting the system components to detect potential failures. To minimize the average asymptotic maintenance cost per time unit, we have to identify the optimal inspection period for the whole system.

Anbar [9] assumes that the inspection frequency for such a policy changes according to the system component failures. At each inspection, he determines the number of failures in the system. From this information, he deduces the next inspection date by computing the average maintenance cost per time unit on an infinite horizon that depends on the observed failures.

Vaurio [169] considers a maintenance policy with regular inspections with a period τ . The component i is inspected at each inspection date $T_k = k.\tau$, $k \in \mathbb{N}$, and is replaced either correctively if it has failed or preventively if the component age reaches $N_{I_i}.\tau$. He develops a method to identify the inspection period τ and the number of inspections N_{I_i} that minimize the average maintenance cost per time unit on an infinite horizon.

3.4.1.3 Opportunistic maintenance

We talk about opportunities when events appear once in a while and are difficult to predict, but offer the possibility to act on the system. If a component fails, opportunistic maintenance enables to combine corrective and preventive maintenance operations at the same time. It then reduces the number of system downtimes and the maintenance costs. Opportunistic maintenance takes also advantage of the system downtimes, caused by external constraints, to perform maintenance operations.

The following cases deal with preventive replacements when a component failure occurs. Ouali et al. [129] consider an opportunistic maintenance strategy for a system composed of n different components. They assume that each component is maintained according to its age. If a component j fails before the scheduled maintenance operation, their opportunistic maintenance policy will correctively maintain the component j and preventively maintain the components $i \neq j$ at the failure time, if their age is greater than $x_i^* + \omega_i$ with $-x_i^* < \omega_i < 0$. They express the average maintenance cost per time unit on an infinite horizon to identify through simulations the optimal replacement age x_i^* and the age $x_i^* + \omega_i$ from which the opportunistic replacement is profitable.

Laggoune et al. [88] suggest an opportunistic maintenance policy for petrochemical plants. They consider a series multi-component system subject to random failures. The components are preventively changed at fixed maintenance intervals. If a component fails between two fixed maintenance dates, it is correctively replaced and other operations may be preventively carried out. They consider all the possible maintenance actions for the non-failing components. For each situation, the total maintenance cost for the system is computing using Monte-Carlo simulations. The minimal cost enables to identify the components to preventively maintain at the failure date.

3.4.2 Dynamic grouping

Dynamic grouping policies also use static maintenance rules but they are completed by short-term information about the components, such as their age, their deterioration level, a change in the operating conditions. The objective is then to adapt the maintenance operations schedule by considering the dynamic monitoring information.

Castanier et al. [37] have developed a condition-based maintenance policy for a system composed of two elementary components. Each component i gradually deteriorates over time. The deterioration for the component i is denoted Z_i . The components are inspected at non-periodic inspection dates. A maintenance decision is proposed to synchronize the inspections and the replacements for the two components to minimize the average maintenance cost per time unit on an infinite horizon. For each component i , two thresholds are defined: a preventive one denoted Z_i^{prev} and an opportunistic one denoted Z_i^{opp} . At each inspection date $T_k, k \in \mathbb{N}$, the components deterioration is checked. For instance, if the deterioration for component 1 exceeds the preventive threshold Z_1^{prev} , it is replaced. Component 2 is also

replaced if and only if its deterioration exceeds the threshold Z_2^{opp} . The next inspection date T_{k+1} is estimated based on the average maintenance cost per time unit on an infinite horizon while integrating the maintenance operations performed at time T_k .

Andréasson [10] develops a dynamic algorithm to group the maintenance operations of the components for an aircraft engine. He classifies the components according to their failure criticality on the system. The critical components have a predetermined and limited lifetime. The « non-critical » components result in a decrease of the system performance. It is assumed that these components lifetimes are known. The system maintenance cost over an infinite horizon is expressed according to the lifetime laws. By minimizing this cost, the components maintenance dates are determined.

In the same context, Dekker et al. [48] and Wildeman et al. [181] propose a dynamic policy to group the maintenance activities on a rolling horizon. In the first place, the optimal maintenance date for each component taken individually is estimated by minimizing the average maintenance cost per time unit on an infinite horizon. A penalty cost function is then built for each component to estimate the additional costs if a maintenance operation is either moved forward or postponed from its optimal date. Based on this information, a dynamic scheduling algorithm is applied to group the maintenance operations on a finite scheduling horizon. From the scheduled maintenance operations on the considered horizon, the objective is to create groups to minimize the total expected maintenance cost over this defined time horizon. These different stages are repeated each time a short-term monitoring information is available. The maintenance operations are then adapted to the structural constraints and to the real usage rate of the components.

In the literature, many studies have sought to develop this maintenance policy. Based on dynamic programming and on the rolling horizon defined in the previous paragraph, Bouvard et al. [32] propose to use this approach for multi-component systems whose deterioration is gradual. The idea is to use information related to the current components state to update the maintenance schedule. In other studies, the methodology based on the rolling horizon was completed to consider time constraints on the maintenance opportunities [52] and constraints related to the number of available maintainers [53]. However, these contributions only deal with series multi-component systems. Other studies enable to adapt the dynamic grouping policy to integrate more complex structures that can combine series and parallel connections [171], [172]. By considering the impact of the system structure, the problem to define the optimal schedule becomes a NP-hard problem. In this context, a genetic algorithm has been developed to schedule the maintenance operations to perform on the defined time horizon. The integration of the structure complexity has then led to the abandonment of the dynamic programming methodology.

Most of the dynamic grouping policies are based on the maintenance opportunity concept [39]. However, due to the way they are built, they do not enable to ensure the system autonomy on a given period. A new dynamic grouping maintenance policy has then been developed based on the Maintenance Free Operating Period (MFOP) [97], [98], [85]. This policy enables to define time periods at the end of each mission when the system can be maintained. By using the available monitoring information and the multi-component system

structure, it is possible to quantify the risk that the system operates until the end of the next mission. According to the risk estimation, the maintenance decision-making process evaluates whether it is necessary to replace or not some components in the system. Contrary to the dynamic grouping policies for which the grouping strategy is performed around the components maintenance dates, in this policy, it is the system that provides the maintenance opportunities.

3.5 Conclusion

This chapter defines the general concepts related to the maintenance activities and gives an overview of the existing maintenance policies for both mono-component and multi-component systems. The objective remains the same no matter the study case: scheduling the maintenance operations at the best moments to minimize a specific criterion.

The available monitoring information on the component(s) plays a dominant role in the choice of the maintenance strategy to adopt. If no monitoring information is available, the maintenance policies are built based on the component(s) properties. These static policies do not make it possible to adapt to the real state of the component(s). Specific systems can be set up to monitor some information. They can give indications on the component(s) state (operating or failed), on the deterioration level or on the operating environment. By integrating that information in the decision-making process, current components state-based policies can be developed. These dynamic policies provide a scheduling method adapted to the real component(s) usage.

In our study, the objective is to adapt at best to the hauler constraints. These constraints mainly lie in the missions definition. Indeed, from one mission to another, the usage conditions can vary a lot and have a significant impact on the vehicle deterioration evolution. In addition, the unavailability periods vary according to the missions to complete. The major difficulty lies in integrating that information to define the maintenance schedule but also to schedule the missions to respect the hauler constraints. The maintenance policy is then seen as a way to control the vehicle health state. A global health indicator is considered for the vehicle. It will enable to make maintenance decisions based on the remaining useful life of the system and adapt them not only to the current vehicle usage but also to its future usage conditions. This predictive maintenance policy is then directly integrated into the missions scheduling. We choose to consider the vehicle as a mono-component system because the focus point of the study is on the fleet aspect. Indeed, to optimize at best the customer productivity, it is necessary to integrate the fleet dimension. But, it is always possible to integrate the multi-component aspect. The easiest way is to assume that the vehicle is composed of components with a series structure. In this case, a failure from one component will still lead to a system failure.

Two parts are highlighted in our thinking. The first one deals with the integrated scheduling for both missions and maintenance operations and the second one adds the fleet dimension to define an integrated schedule for missions and maintenance operations for the whole fleet.

The first one has to be solved before being able to think about the more complex issue brought by the second part.

However, before being able to integrate the predictive maintenance policy to the way to schedule the missions, it is necessary to study how to integrate the usage impact on the health state and the existing optimization methods to obtain a joint schedule.

Optimization methods to jointly schedule maintenance and production

Contents

4.1	Scheduling problem to organize maintenance operations	70
4.2	Models to integrate maintenance and production in the same schedule	71
4.2.1	Interrelated models	72
4.2.2	Integrated models	74
4.2.2.1	Single asset	75
4.2.2.2	Multi assets	78
4.3	Rescheduling methods	82
4.4	Optimization methods	83
4.4.1	Exact methods	84
4.4.2	Approximate methods	85
4.4.3	Multi-agent systems	86
4.4.4	Game theory	87
4.5	Genetic Algorithm	87
4.5.1	Definition	88
4.5.2	Principle	89
4.5.3	Advantages and drawbacks	91
4.6	Conclusion	92

Most production systems rely on optimal and effective scheduling for their different functions. It is usual to plan one function independently from the others, above all because they are treated by different functional teams. We noticed that many haulers independently schedule the maintenance operations and the missions the vehicles have to perform. This practice could lead to sub-optimal schedules insofar as the maintenance schedule is not adapted to the real usage conditions of the vehicle.

Indeed, to manage a production system, we usually look at the whole system. Separating optimal solutions for production and maintenance may not provide an optimal solution for the

whole system. Indeed, when considering the maintenance function, we try to maximize the machines or systems availability, knowing that it will affect the production plan. Similarly, production schedulers may tend to use the systems to their full capacity to meet demand. It may increase the productivity but the availability will decrease due to more systems breakdowns. If we want to reach a global optimal that includes all the functions in the production system, it is necessary to design integrated models to coordinate the functions and minimize the conflicts between these functions. When it comes to manage missions and maintenance operations for vehicles, it becomes essential to jointly optimize the scheduling of both activities to improve the productivity and reduce the failure risks while adapting to the operational constraints.

This chapter aims at giving an overview of the existing studies to jointly schedule the maintenance operations and the production tasks. The objective is to integrate the maintenance model, defined thanks to the different strategies presented in [Chapter 3](#), to optimize the chosen criteria.

In our study, the missions the vehicle have to perform have different severity levels and require different usage conditions due to the roads types, the topography and the delivery loads. This chapter presents the existing optimization methods to tackle the issue of considering the maintenance operations and the production activity to define a joint schedule. It enables to fix more precisely the context of our study and the hypotheses to consider to optimize the joint scheduling method.

Most of the research studies cited in this chapter deal with production systems such as workshop production optimization.

4.1 Scheduling problem to organize maintenance operations

The scheduling problem we are trying to solve is how we can define a joint schedule for both missions and maintenance operations for a vehicle. Integrating the missions into the scheduling problem is like considering the real vehicle usage to be sure to schedule the maintenance operations at the right time according to the vehicle deterioration evolution. As the missions have different usage severity, considering these usage changes enables to model the deterioration evolution more accurately and to define a maintenance schedule with a higher quality. In this case, missions correspond to the production activity.

This issue has been investigated for production systems such as workshops with production machines. The different research studies offer a huge variability when it comes to the problem context, the kind of workshop, the way to consider the system deterioration, the chosen criteria and the optimization method. The general objective is to order the tasks or jobs to do on one or many industrial machines in the production workshops. The workshop scheduling problems are divided into five parts according to the type of workshops that is studied [135]: unique machine, parallel machines, job shop, flow shop and open shop problems. The last one is a bit less documented than the others. Let us define the five different categories that

will be evoked later in the following sections.

For the unique machine and the parallel machines cases, each job has to be processed by one machine. With parallel machines, we can choose among the available ones. Of course, these machines can be either identical or not.

For the standard job-shop problem, we have a list of n jobs of varying processing time that have to be scheduled on m machines. Each job is composed of a set of operations and each operation has to be processed on a specific machine. A job does not need to pass through every machine. It means that each job has its own predetermined route to follow. A common relaxation is the flexible job shop. It means that we can have more than one machine that can process a specific operation. On the contrary, for the flow shop problem, the m machines are in series and all jobs have to follow the same route and be processed by all the machines. They have to be processed first on machine 1, then on machine 2, and so on. One of the variant is to consider that, instead of having m machines in series, we have c stages in series. At each stage, m machines in parallel are available. It is called the flexible flow shop. The last category is the open shop. Each job has to be processed again on each one of the m machines. However, some of these processing times may be zero. There are no restrictions with regard to the routing of each job through the machine environment. The scheduler is allowed to determine a route for each job and different jobs may have different routes.

Jointly scheduling the maintenance operations and the missions or the jobs is a NP-hard problem as it is the case for joint maintenance and production scheduling problems in workshops [17], [23], [92], [96], [109], [117], [176]. Indeed, it is a combinatorial problem that can be solved with enumeration methods for small-size problems. However, when the problem size increases, we reach the combinatorial explosion very fast. In these cases, other methods have to be used instead of exact methods to obtain a suitable solution.

There are different models to jointly consider maintenance operations and production. We can either start by scheduling one of the two activities and consider this schedule as a constraint to plan the other activity. Or, we can schedule them simultaneously. The first model class groups the interrelated models or sequential models while the second class deals with the integrated models [73].

4.2 Models to integrate maintenance and production in the same schedule

After defining the scheduling problem, we depict the existing models developed in the literature to coordinate the scheduling of maintenance and production activities.

There are two levels of scheduling coordination. The first one groups what we call the interrelated models. They are models for which the decision variables only concerned the original function we try to optimize. It means that they either consider the production schedule or the maintenance schedule as a constraint to schedule the other activity. The

second level deals with the integrated models. In these models, the decision variables are for both functions. They are usually not easy to solve because of their multi-objective nature.

4.2.1 Interrelated models

Interrelated models consider optimizing a certain function of the production system by considering the requirements of another function as a strong constraint [73]. For instance, models exist for scheduling jobs while considering machine unavailability due to maintenance operations. The other way around is also studied where maintenance operations are scheduled using the production schedule as a constraint [22]. These models have been created due to the fact that planners may give a higher priority to one function and plan for that solely. The output plan will then be used as an input to schedule the second in priority function. This scheduling situation is seen as a coordination between the different functions rather than real integration. It explains why we talk about interrelated models rather than integrated models.

When it comes to study production systems, most of the interrelated models are based on the assumption that the machines are not continuously available. For the single machine scheduling problem, some research studies only consider one unavailability period due to maintenance operations. Schmidt [153] presents an review related to deterministic scheduling problems where machines are not continuously available. Sadfi et al. [147] study the single machine total completion scheduling problem subject to a period of maintenance. To solve the problem, they use a heuristic method based on a post-optimization of the solution obtained when applying the Shortest Processing Time algorithm. For Low et al. [108], the objective is to minimize the makespan in the system while considering an unavailability period. They consider a simple linear deterioration model for the machine deterioration evolution to be in a deterministic study case. Some heuristic algorithms, based on the bin packing problem, are then developed to solve the problem. They study both the preemptive and non-preemptive assumptions for the jobs.

Other studies consider the maintenance task as a special job that should be carried out periodically with a predetermined interval. The schedule then contains several maintenance periods. Raza et al. [138] address the maintenance and jobs joint scheduling problem to minimize the total earliness and tardiness with a common due date. They propose three algorithms, a constructive heuristic and two meta-heuristic methods based on the properties of an optimal schedule. The machine has a maximum continuous operation time period that can not be exceeded. After this period, maintenance operations are carried out during a fixed period. It enables to define jobs batches separated by maintenance operations. Liao and Chen [101] try to minimize the maximum tardiness with periodic maintenance and non-resumable jobs. Several maintenance periods are under consideration where each maintenance operation is required after a periodic time interval. Chen [38] works also on his own on the single-machine scheduling problem with periodic maintenance and non-resumable jobs. Based on the Moores algorithm, he develops an effective heuristic method to obtain a near-optimal schedule for the problem that minimizes the number of tardy jobs. Low et al. [109] study the single-machine scheduling problem to minimize the makespan but they consider that the

machine has to be maintained, either after a periodic time interval or to change tools after a fixed amount of processed jobs.

Considering unavailability constraints for the multiple machine scheduling problem has also been investigated. Schmidt [152] studies the parallel machine preemptive scheduling problem, for which each job has a deadline and each machine has different availability intervals. Lee [93] tries to minimize the makespan for parallel machines where machines may not be all available at the same time. To solve this problem, he proposes a modified Longest Processing Time algorithm. Ho and Wong [74] also study the makespan minimization on parallel machines, and propose a Two-Machine Optimal (TMO) scheduling algorithm. Liao et al. [102] are also interested in solving the two-parallel machine problem to minimize the makespan and introduce a fixed and known in advance unavailable time period for one of the two machines. Lin and Liao [103] complete the previous study on the two parallel machine problem. But, they take into account that each machine has a fixed and known unavailable time period. Lee and Liman [95] consider a variant of the two-parallel machine problem to minimize the sum of job completion times. They assume that one of the machines is available for a specified time period after which it can no longer process any job. Mosheiov [116] extends this work by considering that all the machines are shut down at some point. Lee and Chen [94] investigate minimizing the total completion time for m parallel machines, where each machine should be maintained at least once in the planning horizon. They study two versions of the problem. The first one considers that more than one machine can be maintained simultaneously while, for the second case, only one machine at a time can be maintained.

The flow shop scheduling problem with unavailability constraint has also been solved by using interrelated models. Many research studies work on the two-machine flowshop scheduling problem. The difference comes from the considered hypotheses when defining the schedule. They can assume that the preemption is allowed. It means that a job can be stopped during its execution to perform a maintenance operation. After the maintenance period, the job can restart from where it stopped. It is the resumable job case. If it has to restart from scratch, we are in the non-resumable case. Allouani et al. [8] consider the two-machine flow shop problem to minimize the makespan. They focus on the non-resumable case and assume that one of the two machines must be maintained once during the first T time periods of the schedule. They study the optimal solutions when using Johnson's algorithm. Blazewicz et al. [28] study the same problem but consider unavailability intervals for both machines and allow preemption as well as resumable jobs. They propose heuristic methods to minimize the makespan. For Liao and Tsai [105], the preventive maintenance intervals depend on the number of finished jobs. They develop different heuristic methods to minimize the makespan and employ a Branch and Bound algorithm to find the optimal solution and evaluate the performances of their heuristic algorithms. Preemption is not allowed in their study as well as for Yang et al. [187] and Kubzin and Strusevich [84]. Kubzin and Strusevich [84] consider that only one machine has to be maintained once, and that the maintenance intervals have varying length according to the machine state. They also add an hypothesis on the way to process jobs: the no-wait process. It means that the processing of a job operation on machine B must start exactly when the operation is completed on machine A. A polynomial-time approximation scheme is developed to minimize the makespan. Kubiak et al. [83] propose a

Branch and Bound algorithm to minimize the makespan for the two-machine flow shop with periods of limited availability and with resumable jobs.

Complementary studies consider flow shop composed of m machines. Each job visits every machine. Aggoune [4] assumes that each machine has several unavailability periods due to preventive maintenance. The operations composing each job are non-preemptive. It means that they cannot be stopped once started. He considers this hypothesis as well as the unavailability constraint to schedule the jobs to minimize the makespan. He shows the interest of using time window for maintenance periods rather than fixed intervals. Aggoune and Portmann [5] study the flow shop scheduling problem with limited availability and non-preemptive jobs. They propose a heuristic approach to approximately solve the problem that consists in scheduling the jobs two by two according to an input sequence and using a polynomial algorithm. It is actually an extension of the two-machine scheduling problem approach.

Some studies add a complexity to the workshop structure by considering hybrid flow shop scheduling problems. Each stage of the flow shop is composed of several parallel machines. Allouani and Artiba [7] consider a two-stage hybrid flow shop with non-preemptive jobs. The first stage is composed of one machine while the second is composed of m parallel machines. They aim at minimizing the makespan with a deterministic unavailability constraint, assuming that there is, at most, one unavailability period per machine. The objective of Gholami et al. [68] is also to minimize the makespan of a hybrid flow shop. However, they consider sequence-dependent setup times, non-preemptive jobs and assume that machines can suffer stochastic breakdowns. They develop a method to incorporate simulations into a genetic algorithm to solve the scheduling problem.

When applying interrelated models, either the maintenance function, or the production function, has a higher priority above the other, as we use the output schedule of one activity as a constraint to schedule the second activity. In our study context, the Volvo Group is in charge of the maintenance schedule and delivers it when a vehicle is purchased by a customer. The production schedule is then defined by the customer, knowing that the customer sometimes complains about the maintenance schedule that forces him to postpone some missions. From this statement, the two activities should not be dissociated when defining the global schedule for a vehicle. In addition, none of them seems to take priority over the other. To find a global optimal schedule, it may be relevant to apply an integrated model rather than an interrelated model.

4.2.2 Integrated models

A production system needs more than just coordination, offered by interrelated models, to increase productivity and reduce costs. It justifies the development of integrated models to consider the objectives to reach for the maintenance function and for the production function simultaneously. Indeed, production planning is very dependent on the system conditions. Breakdowns may disturb the production activity and cause delay in the schedules. These

statements led the researchers to spot out the need to integrate production and maintenance schedules to generate expected cost savings and better use the resources. The maintenance schedule might delay the production schedule, but it will reduce the expected number of failures on the system. It then becomes a win-win situation. Moreover, no priority is given to any activity as the objectives of both maintenance and production are considered at the same level.

The studies found in the literature can be divided into two parts. The first one focuses on the joint scheduling problem for a single system, while the second one is interested in the joint scheduling problem for a fleet or a group of systems. In our research work, as the customer can be the owner of only one vehicle or a fleet of vehicles, studying both cases is relevant.

4.2.2.1 Single asset

Several research works have proposed methods to solve the integrated problem for production scheduling and maintenance planning for a single system. They all consider different hypotheses, either regarding the production model, the maintenance strategy, the effect of the system deterioration or the considered objective functions to optimize. The contributions are split into two parts: the ones considering a stochastic framework and the ones considering a deterministic framework to solve the integrated problem on a single system.

Cassady and Kutanoglu [35] propose an integrated model that coordinates the preventive maintenance decisions with single-machine scheduling decisions to minimize the total expected weighted completion time of jobs. They assume that the time to failure for the machine is governed by a Weibull probability distribution. When the machine fails, minimal maintenance is performed, i.e. it is restored to an operating condition without altering its age. On the contrary, preventive maintenance operations restore the machine to an « As Good As New » condition and its age goes back to zero. As the objective is to maximize the machine availability, an age-based preventive maintenance policy is applied to determine the optimal date for preventive maintenance. As preventive maintenance is perfect, the machine maintenance can be modelled as a renewal process for which the renewal points correspond to the initiation of machine operation and the end of each preventive maintenance activity. The failure occurrences during each cycle of the renewal process are modelled using a non-homogeneous Poisson process as repair is minimal. These models enable to define the expression of the steady-state availability of the machine. The integrated problem is complicated insofar as the completion time of each job is stochastic. Indeed, the machine may fail during each job and the preventive maintenance decisions change the stochastic process governing machine failure. The integrated problem is solved using total enumeration. The first step is to enumerate all the feasible job sequences and then identify the optimal set of preventive maintenance decisions for each feasible job sequence. A comparison is drawn between the integrated solution and the solution obtained from solving the preventive maintenance planning and job scheduling problems independently. Integrating the two decision-making process results in average savings of 2% and occasional savings that can reach 20%. Kuo and Chang [86] are interested in finding the optimal integrated production schedule and maintenance planning to

minimize the total tardiness of a fixed number of jobs, processed on a single machine under a cumulative damage process. They also investigate how the optimal preventive maintenance plan interacts with the optimal production one. They consider that all jobs are available at the beginning of the schedule definition and that no future job will arrive. The scheduling problem is then a static one. Actually, their problem definition is quite similar to the one made by Cassady and Kutanoglu [35], but the difference is that they solve it analytically. Sortrakul et al. [157] propose different genetic algorithm heuristics to solve the integrated problem for a single machine introduced by Cassady and Kutanoglu. They aim at minimizing the total weighted completion time for all the jobs while considering that the machine time to failure follows a Weibull distribution. Minimizing this function means finding the best job sequence and the right moments to perform the preventive maintenance operations. When the machine fails, they also perform a minimal repair and the preventive maintenance operations restore the system to an as good as new state. They draw a comparison with the algorithm developed by Cassady for different sizes of the problem.

The previous papers are focused on the jobs completion time optimization. However, some contributions propose solutions for the integrated problem to optimize the costs incurred by production and maintenance activities. Li et al. [100] describe an integrated model for production and maintenance scheduling to minimize the total cost of the manufacturing system. According to them, using an integrated model to schedule both activities is more relevant as preventive maintenance is performed according to the equipment status in the production process. Moreover, it ensures the effective availability of the equipment and reduces the machine downtime losses. They assume that the maintenance activity is imperfect, governed by an improvement factor, and failure occurrences lead to minimal repairs. They assume that the failure rate follows a Weibull distribution and use genetic algorithms to find the optimal schedule. They compare the integrated model with a strategy treating each activity scheduling independently. The integrated model enables to generate about 12% savings on the study case. However, they only consider steady-state availability in this model without considering any more complex constraints based on reliability or failure risk. Kiani and Taghipour [82] also propose a method to find the optimal sequence of jobs and the jobs before which preventive replacements should be conducted. They want to minimize the expected costs incurred over makespan, i.e over the completion time for the last job in the sequence. These costs include tardiness penalties as well as corrective and preventive maintenance costs. They consider a single dominant failure mode and model the system failure with the concept of delay time model (DTM). DTM is a two-stage failure process in which an initial defect will eventually lead to failure if it is left unattended. The elapsed time between a defect occurrence and the failure, in the absence of inspection, is called delay time and provides an opportunity window to inspect the system and fix the defect. The defect arrival in the system is assumed to follow a non-homogeneous Poisson process with a power law intensity function and the delay time follows a Weibull distribution. The integrated problem is solved by enumerating all the possible schedule solutions and evaluating them to find the optimal one. Note that all the jobs have a common due date. Fitouhi and Nourelfath [60] deal with the problem of integrating noncyclical preventive maintenance production planning for a single machine. A set of products is given and must be produced in lots during a specified finite planning horizon. The maintenance policy suggests possible preventive maintenance replacements at

the beginning of each production planning period and minimal repair at machine failure. The proposed model determines simultaneously the optimal production plan and the instants for the preventive maintenance action. The objective is to minimize the preventive and corrective maintenance costs, setup costs, holding costs, backorder costs and production costs, while satisfying the demand for all products over the horizon. The problem is solved using mixed integer linear programming to determine the integrated production and maintenance plans. Their study shows that the integration of maintenance and production planning can reduce the total costs and the removal of the periodicity constraint is directly affected by the demand fluctuation and also helps to reduce the total production and maintenance costs.

The production and maintenance integrated problem can also be defined as multi-objective problem with an objective function related to the completion time for all the jobs and a cost-based objective function. Liao et al. [106] define a single-machine optimization model of production and preventive maintenance for group production to minimize the total completion time of jobs and the maintenance costs. Each group is composed of several jobs that contain some identical parts to be produced. A hybrid maintenance strategy is adopted. Preventive maintenance operations are performed to renew the machine while considering failure rate threshold. Note that the failure rate follows a Weibull distribution. For corrective maintenance, minimal repair is performed based on the machine's age. The model also includes the machine deteriorating effect, based on the machine's age, as well as the learning and forgetting effects. A genetic algorithm based method is applied to solve the integrated problem and shows that this model enables to avoid excessive and insufficient maintenance. Yulan et al. [189] also study the integrated scheduling problem but consider five objective functions. They aim at minimizing the makespan, the total weighted completion time of jobs, the total weighted tardiness, the maintenance costs and maximize the machine availability. The first three objectives enable to find the optimal sequence of jobs. Minimizing the makespan and maximizing the machine availability are quite similar objectives while the others are always in conflict. But, integrating both activities in the schedule is necessary because the completion time of each job depends on several factors. These factors are the machine reliability before processing the job, the preventive maintenance decisions and the time to complete them, the number of machine failures during the job, the time to complete maintenance after failures, the completion time of the previous processed jobs and the job processing time. A multi-objective genetic algorithm is then developed to solve this problem. The total weighted percent deviation, that represents the preferences of the objectives and the deviations of the solutions, is proposed to help decision-makers select the best solution among the near-Pareto optimal solutions obtained by the algorithm.

Other research works study the integrated problem but assume that the system evolves in a deterministic environment. Yang et al. [188] consider single-machine due-window assignment and scheduling with job-dependent ageing effects and deteriorating maintenance. All the jobs have a common due-window and the objective is to jointly find the optimal time to perform the only one maintenance operation on the time horizon, the optimal location and size of the due-window, and the optimal job sequence to minimize the total earliness, tardiness and due-window related costs. The processing time for each job only depends on the job ageing factor. The maintenance operation is assumed to be perfect and its duration is a

linear function of the time depending on its starting time and a maintenance factor. All the factors are fixed and known values. That is why, the framework is deterministic. Ladj et al. [87] aim at scheduling several jobs for production and predictive perfect maintenance interventions to minimize the maintenance costs. The planning horizon can be divided into multiple production cycles separated by predictive maintenance operations. It means that jobs are grouped into blocks separated by predictive maintenance operations while respecting the constraint on the machine maximum capacity Δ . This capacity corresponds to the maximum deterioration value leading to the system breakdown. The maintenance cost is calculated based on the cumulated deterioration of the assigned jobs. However, it is assumed that a degradation value is assigned to each job and is known since the beginning of the planning process. The deterioration evolution is then clearly deterministic. In this framework, the problem definition looks a lot like the bin packing problem. To solve it, a hybrid algorithm based on genetic algorithm and artificial immune systems is developed. This heuristic method enables to search for the optimal solutions and the created immune operators, based on the artificial immune systems, aim at avoiding the premature convergence of the algorithm.

The presented methodologies to solve the integrated problem for a single system show that the defined framework varies a lot according to the hypotheses considered for the system, the size of the problem and the complexity of the resulting planning. Exact or heuristic methods can be applied to solve the problem, considering either a deterministic or a stochastic framework. The optimization function depends also a lot on the objective we are trying to reach with the integrated scheduling problem. It can either be based on the maintenance costs, or on the time necessary to complete all the tasks or a mix of both objectives. This variety is also observed when we study the existing methodologies to solve the integrated problem for a fleet of systems.

4.2.2.2 Multi assets

Extending the integrated scheduling approach for production and maintenance from a single system to a fleet composed of several systems adds a complexity level to the joint scheduling problem. Indeed, the problem is not just to order the production and maintenance activities but also to assign the different tasks to the different vehicles according to their configuration and health state. Less research works are available regarding the joint scheduling problem for fleets or the existing ones consider simple maintenance policies.

Indeed, several articles consider a periodic maintenance policy to deal with the joint scheduling problem for maintenance and production for a group of systems. Benbouzid-Sitayed et al. [23] study the joint scheduling problem for a flow shop. They propose a heuristic (Nawaz-Enscore-Ham heuristic) and two meta-heuristic (Genetic Algorithm and Tabu Search) methods to optimize the production and maintenance schedule with both production and maintenance criteria. They assume that the machines do not fail during the time horizon and only describe the jobs to complete by their processing time. The objective is to minimize the makespan (time when the last scheduled task is over) and respect the maintenance periods because it influences a lot the constraints on the production system.

Avoiding earliness or tardiness for the maintenance operations is then essential. Each machine is periodically maintained at known intervals but the periodicity of these operations can still vary in a tolerance interval. However, they all have the same durations. In addition, all the machines complete the same sequence of jobs. The comparison between the three methods on numerical examples show that the genetic algorithm-based method gives the best results. As only the jobs processing times are taken into account for the schedule definition, the problem study is deterministic. Yalaoui et al. [184] treat the dual objective problem of production planning and maintenance with a mixed integer linear program. Their objective is to minimize the total costs i.e. the production and maintenance costs. To do so, they consider the deterioration of the different production lines as a reduction of their capacities according to the time evolution. The time horizon is divided into cycles of production. The maintenance operations restore the lines capacities at their maximum and all the preventive maintenance operations are performed at the beginning of each cycle. Then, they also study the integrated scheduling of production and cyclic preventive maintenance. Berrichi et al. [27] develop an algorithm based on multi-objective ant colony optimization for parallel machines. They want to assess the near-optimal Pareto solutions offering the best assignment of production tasks to machines and preventive maintenance periods. It is a bi-objective approach to find a trade-off between production and maintenance objectives. They consider the makespan as performance measure for the production optimization and the unavailability of the complete system with all the machines to determine the periodic perfect preventive maintenance dates for each machine. They use two ant colonies that collaborate together to reach the objectives: one for the maintenance optimization and one for the production optimization. They also model the time to failure using an exponential probability distribution with a failure rate λ and a repair rate μ . Berrichi and Yalaoui [26] also study the joint scheduling problem with similar hypotheses as Berrichi et al. [27]. They consider the total tardiness and the unavailability of the production system composed of parallel machines. The different is that they develop an algorithm using Pareto ant colony optimization (P-ACO). However, P-ACO needs a local search to obtain near-optimal and well-distributed Pareto fronts in reasonable time. The two colonies of ants, one for maintenance optimization and one for production optimization, cooperate through suitable and efficient heuristic information and the acquired knowledge is shared at the end of each cycle. Nourelfath et al. [126] integrate preventive maintenance with tactical production planning in multi-state systems to minimize the total expected cost. To replace the context, a set of products is given and must be produced in lots on a multi-state production system during a specified finite planning horizon. Preventive maintenance or unplanned corrective maintenance can be performed on each machine, also called component, of the multi-state system. The defined maintenance policy considers periodic preventive replacements for the components and minimal repair on the failed ones. The objective is to determine an integrated lot-sizing and a preventive maintenance strategy for the system that will minimize the preventive and corrective maintenance costs, the setup costs, holding costs, backorder costs and production costs, while satisfying the demand for all the products over the entire horizon. They model the production system as a multi-state system composed of machines and each machine has two states: either good or failed. These machines are assumed to be independent economically, stochastically and structurally. Economic independence implies that the cost of joint maintenance of a group of components is equal to the

total cost of individual maintenance of these components. Stochastic independence means that the condition of components does not influence the lifetime distribution of other components. Structural independence applies if each component structurally forms an entity that is not further subdivided for a reliability study. The time horizon is divided into periods and a certain demand has to be satisfied at the end of each period. They consider periodic perfect maintenance operations that bring back the machine age to zero and minimal repair when failures occur. Note that minimal repair does not alter the machine age and enables to model the failure occurrences using a non-homogeneous Poisson process. The production planning part corresponds to a multi-product capacitated lot-sizing problem. Decisions involve the identification of items quantities (lot sizes) to be produced in each period. To estimate the production costs, it is necessary to estimate the average production rate of the system for each period. The first stage is to evaluate the average availability of each component per period. As the periods are long enough, they can consider the steady-state availability for each component. Once it is calculated for each component on each period, an appropriate evaluation method can be applied to estimate the system average production rate on each period while considering the system configuration (series, parallel, series-parallel,...). They use mixed-integer non linear programming to describe the problem and propose two methods to solve it. The first one is an exhaustive evaluation of all the preventive maintenance (PM) solution. For each PM solution, the problem is solved as a multi-product capacitated lot-sizing problem. It works for small-size problems but, for bigger-size problems, the number of mixed integer programs to solve becomes too large. Then, they develop an alternative solving method based on a genetic algorithm that optimizes both production and maintenance costs. The genetic algorithm enables to save some significant computation time. Actually, in the numerical example considering 10 machines and 5 periods, the genetic algorithm only needs about seven minutes to reach the optimal solution while the exhaustive enumeration method needs more than six days, as there are 10^5 PM solutions to evaluate.

However, some studies do not consider a periodic preventive maintenance strategy to solve the joint scheduling problem. Xiao et al. [182] develop a joint optimization model connecting group preventive maintenance with production scheduling, applied on a series system where preventive maintenance on any machine leads to the unavailability of all machines. They assume that the preventive maintenance policy is perfect and minimal repair is performed in case of failures. They aim at minimizing the total system costs induced by production, preventive and corrective maintenance as well as tardiness. A random-keys genetic algorithm is developed to determine the optimal sequence of assigned jobs and the optimal group preventive maintenance interval. Note that random keys only refer to the fact that each solution is encoded as a vector of random numbers belonging to the interval $[0; 1]$. Feng et al. [57] study the problem of different families of jobs that have to be processed on different machines for a flowshop. Each job has to be processed on each machine with a fixed order. Their objective is to find the optimal production sequence of job families, the optimal sequence of individual jobs in each family and the optimal preventive maintenance decisions before each job for each machine to minimize the jobs tardiness costs and the preventive and corrective maintenance costs. Note that all families and jobs order are the same for each machine. They assume that the preventive maintenance operations are imperfect and enable to reduce the cumulative failure risk. They use a genetic algorithm to solve the problem and combine it

with elitism strategy and local search to improve the algorithm performance. Fitouhi and Noureldath [61] propose an update of their previous paper [126] by studying the integration of noncyclical preventive maintenance with tactical production planning in multi-state systems. They consider the same hypotheses, except for the preventive maintenance strategy. The time horizon for production is divided into T periods of fixed length and each period is also divided into S equal sub-periods called maintenance planning periods. For each machine, there are $T \times S$ decision variables corresponding to all the possible preventive replacement actions that can be carried out at the beginning of each maintenance planning period. An age matrix is obtained based on the maintenance policy matrix to represent the effective age of each machine at the beginning of each maintenance planning period. Then, a matrix based methodology is applied to estimate the model parameters, such as the system availability and the general capacity. The defined model is solved using either the exhaustive search (ES) method or the Simulated Annealing (SA) algorithm. The ES method enables to reach the optimal solution, but the computation time is long because of the high number of combinations. It can only be used for small production systems. On the contrary, the computation time is significantly reduced when using the SA algorithm but it does not always converge towards the optimal solution. Nevertheless, the deviation is quite small when it does not converge so the obtained solution remains close to the optimal one. This work manages to show that the integration of acyclical maintenance and production planning helps to reduce the total production and maintenance costs. Da et al. [46] discuss the integrated problem on uniform parallel machines with different maintenance cost, failure rate and processing rate to minimize the makespan and the maintenance costs. They assume that each machine deterioration follows a Weibull distribution and that the machine ageing affects the processing time of the jobs. To improve the machine health status, an imperfect maintenance policy is applied. A failure rate adjustment factor and an effective age improvement factor for each machine are defined and used to estimate the impact of each maintenance operation. Note that a preventive maintenance intervention on a machine leads to the unavailability of all the machines. The maintenance strategy is flexible with dynamic preventive maintenance intervals and replacement timing. The genetic algorithm NSGA-II is applied in this case, insofar as it has three major improvements with respect to NSGA: using quick non-dominated sorting algorithm to reduce computational complexity, introducing elitism strategy to enhance optimization accuracy and adopting density estimation and congestion comparison operators to extend individuals in quasi-Pareto domain to the entire Pareto domain and guarantee population diversity. This algorithm enables to find out the optimal job sequence, maintenance schedule, and machine assignment simultaneously.

It is sometimes difficult to follow the advances in the asset management area insofar as researchers employ a different terminology to refer to their specific problem. Most of the time, the fleet are composed of identical assets. Another form of fleet is defined as a group of vehicles or machines that are not necessarily identical, but share mutual technical features and work under similar conditions. Sriram and Haghani [158] consider a fleet of aircrafts as a pool from which any plane can be assigned to any origin-destination route. Then, the maintenance planning is scheduled considering different intervention levels. Petchrompo and Parlikad [132] propose in their review a classification and a definition of multi-unit systems based on the existing literature about asset management. This classification is based on

the diversity of assets and the intervention options when it comes to maintenance. They also study the different dependencies for multi-unit systems, the different types of problems regarding fleet management and the methods applied to solve them.

The joint scheduling problem for a group of systems also starts to be studied through the articles published in the literature. However, they simplify the issue to reduce its complexity and be able to tackle it. The maintenance policies are often periodic ones and the health state of the systems is not often considered as such to make decisions regarding the maintenance and production scheduling. We can note that meta-heuristic methods, such as genetic algorithm, are applied in most cases, above all for large-size problems. Nevertheless, the developed methodology enables to define a joint schedule for both production and maintenance. But, it does not consider the opportunities offered by the schedule execution to update it, based on the potential collected monitoring information. It would then also be relevant to review the existing methods to reschedule the initial joint planning.

4.3 Rescheduling methods

The existing approaches to model production and maintenance at the same time have been described in the previous part. However, all these approaches are static, i.e. they do not offer the opportunity to change or update the schedule if monitoring information is available. This section focuses on the rescheduling research topic and the different parameters to define when considering rescheduling.

Rescheduling consists in updating an existing schedule in response to disruptions or other changes [170]. It includes the arrival of new jobs, machine failures and machine repairs. Vieira et al. [170] propose an overview of the appropriate definitions for most applications of rescheduling manufacturing systems and describe a framework for understanding rescheduling strategies, policies and methods. They focus on the rescheduling of production activities but adding the maintenance activity does not change the principle of rescheduling and the points to focus on when defining a rescheduling strategy. They define the scope of rescheduling research according to three axes: the rescheduling environment, the rescheduling strategy and the rescheduling method. The environment identifies the set of jobs the schedule should include. It can either be static with a finite set of jobs or dynamic. The rescheduling strategy describes whether or not schedules are generated. There are two types of strategies: the dynamic and the predictive-reactive strategies. For a dynamic strategy, there is no schedule generation but dispatching rules, or a control strategy, to dispatch the tasks. For a predictive-reactive strategy, a schedule is generated and updated to consider the disruptions occurrences. Note that the predictive-reactive strategies are the most commonly used in practice. When defining a strategy, a rescheduling policy is necessary to specify when and how rescheduling is done. It mainly specifies the events that trigger rescheduling. Finally, the rescheduling methods are either to generate a schedule or to repair it. They are then only applicable for the predictive-reactive strategies. Aytug et al. [15] also review the literature on existing production schedules in the presence of unforeseen disruptions on the shop floor. When

studying the existing research on scheduling with uncertainty, they consider three categories based on the problem formulation: the completely reactive approaches, the robust scheduling approaches and the predictive-reactive approaches. The completely reactive approaches are based on dispatching rules. The robust ones focus on creating a schedule that minimizes the disruptions effects on the performance when implemented. When considering predictive-reactive scheduling, two main questions are raised: when to reschedule and what actions are done. The first answer is that rescheduling is needed when an event has an impact significant enough to justify the generation of a new schedule. The underlying need is then to estimate if rescheduling is worth it or if a schedule update will be too expensive in terms of resources, costs and changes. Then, the choice of the strategy to adopt to trigger rescheduling has to be made. Different policies exist, such as continuous rescheduling that reschedules at each event, periodic rescheduling or event-driven rescheduling that initiates rescheduling when an event can cause serious disruption. They all have their advantages and drawbacks. For instance, continuous rescheduling runs the risk of initiating rescheduling for events that are not significant enough. Periodic rescheduling leads to ignore the occurring events between two rescheduling points. Hence, a combination of periodic and event-driven approaches can be attractive insofar as it considers significant disruptions.

However, the rescheduling problem is mostly studied without considering the maintenance activities. For instance, Hoogeveen et al. [76] consider the rescheduling problem for a single machine to schedule sets of jobs to reduce the production costs and limit disruptions by avoiding too many schedule updates. They allow the original schedule to be updated but each disruption incurs a certain cost. This cost is estimated based on three indicators: the absolute positional disruption, i.e. the absolute difference between the job position in the initial schedule and in the new one, the positional disruption and the absolute completion time disruption. The few research papers investigating the rescheduling problem while considering the maintenance activities often apply basic maintenance strategy. Wang et al. [173] study the rescheduling problem in response to the arrival of new jobs for a single machine where preventive maintenance should be determined. Their objective is to optimize both the total completion time to do the jobs as well as the maintenance and production costs. The processing time of each job is affected by the deterioration, and the allocation of more resources on a job can reduced its processing time. However, only one maintenance slot has to be scheduled. Its effect on the machine deterioration varies according to its duration.

4.4 Optimization methods

Solving the joint production and maintenance scheduling problems, for either a single system or a fleet of systems, requires optimization methods to reach a feasible solution. This solution can be optimal or approximate. Several methods have been developed in operational research to solve complex optimization problems. Fitouhi [59], inspired by the overviews of Floudas and Gounaris [62] and Nahas [119] on the research advances regarding global optimization problems, classifies the optimization problems in two clusters according to the quality and the optimality of the solution: the exact or optimal methods and the approximate or incomplete

methods.

This section starts with a presentation of the existing optimization methods to solve the joint scheduling problems based on the ones mentioned in [Section 4.2](#). Then, two special parts are focused on two more complex existing methodologies. They are presented to give a complete overview of the optimization methods but they are not our focus point for the study led in the following chapters.

4.4.1 Exact methods

The exact methods enable to determine accurately the optimal solution of the problem. However, they are only applied for small-size problems insofar as the computation time increases exponentially with the problem size. Despite the current available technologies, the computation time remains long and sometimes solving the problem with these methods may be impossible for large size problems. There are four different exact optimization methods used to solve scheduling problems.

- **Exhaustive Search (ES) method or enumeration method:** This method consists in evaluating all the feasible solutions for the problem and their corresponding objective function value to identify the optimal solution.
- **Mixed-Integer Linear Programming (MILP):** This method develops mathematical linear programs under constraints and with decision variables that are integer and/ or real. The MILP problems resolution is mostly based on the Branch and Bound method and the decomposition of Benders [\[59\]](#). The first approach implies an enumeration of the integer sub-problems using the relaxation technique. The Bender decomposition is normally used for problems having binary variables. It consists of a decomposition setting certain decision variables in a main sub-problem.
- **Mixed-Integer Non Linear Programming (MINLP):** These problems suggest mathematical non linear models with decision variables that are either integer and/ or real. Different resolution methods have been developed in the literature. They are generally based on the problem features such as its convexity, concavity or continuity [\[62\]](#).
- **Branch and Bound algorithm:** It consists in doing an implicit enumeration by dividing the problem in sub-problems and evaluating them using a relaxation (continuous of Lagrangian mainly) until having only easy problems to solve or for which we know for sure that they cannot contain the optimal solution [\[119\]](#). The set of candidate solutions is forming a rooted tree with the full set at the root. The algorithm explores branches of this tree, that represent subsets of the solution set. Before enumerating the candidate solutions of a branch, the branch is checked against upper and lower estimated bounds on the optimal solution, and is discarded if it cannot produce a better solution than the best one found so far by the algorithm. This algorithm depends on efficient estimation

of the lower and upper bounds of regions/branches of the search space. If no bounds are available, the algorithm degenerates to an exhaustive search.

4.4.2 Approximate methods

The approximate methods are an alternative to handle the hard, combinatorial or large-size optimization problems. These methods enable to reduce significantly the computation time but do not ensure to reach the optimal solution. The meta-heuristic or heuristic methods presented in the literature allow us to find sub-optimal solutions for problems that are impossible to solve with exact methods. There are five different methods mentioned in the previous sections.

- **NEH heuristic:** This heuristic method consists in finding an initial solution and improving it by applying iterative permutations until the best solution is identified [23]. The algorithm is based on the assumption that a task with a high execution time has priority over a task with a lower one. To begin with, tasks are sorted in descending order of their execution time. Then, the algorithm selects the two longest tasks to perform and orders them to minimize the makespan. Next, it selects the longest unordered task and inserts it in the previous order to minimize the makespan. The insertion step is repeated until all tasks are ordered.
- **Tabu Search:** It is a memory based meta-heuristic method. Its aim is to visit the solution space without falling into a local optimum and without cycling in the algorithm [4]. It is an iterative approach that starts from an initial solution (feasible or not) and tries to improve it until a stopping criterion is reached. At each iteration, the current solution neighbourhood is explored and the next solution is obtained by finding the right move minimizing the objective function. Note that this operation may lead to increase the objective function value. It is the way to get out of a local optimum. To avoid circling around, it is important to have a memory mechanism that forbids to come back to the last explored solutions. The previous moves are then stored in a tabu list of size S . S is a parameter of the algorithm. The move remains in the tabu list during S iterations. Information given by the tabu list is used by the tabu restriction to classify some moves as forbidden. But a tabu move status is not absolute and can be eliminated if certain criteria are satisfied.
- **Simulated Annealing algorithm:** This method enables to avoid convergence problems due to local minimum contrary to other meta-heuristic methods. It is similar to the process used by steelworkers to obtain a well-ordered state of minimal energy metals. The process consists in increasing the material temperature and decreasing it slowly to avoid the metastable structures. The meta-heuristic method is based on the Metropolis algorithm by introducing the notion of fictive temperature. Starting from a given configuration, the system goes through a series of elementary modifications that are accepted if they reduce the objective function or if they have a probability equal to $\exp(\Delta E/T)$ [59].

- **Ant Colony:** The ant colony optimization algorithm is based on the collective deposition and tracking behaviours observed in ant colonies. The principle is explained as follows. A colony of agents, the ants, communicates indirectly via dynamic modifications of its environment, pheromone tracks, and thus builds a solution to a problem based on its collective experience. Ants use pheromone tracks to mark their path [27]. The higher the amount of pheromones is present on a route, the more likely the ants will follow this trail. Hence, this path has a higher probability to be taken.
- **Genetic Algorithm:** The genetic algorithm is a meta-heuristic method that belongs to the evolutionary methods class. It is based on the natural selection process developed by Darwin. The algorithm is based on a loop composed of a sequence of selection and crossover stages. The individuals represent the solutions and form a population. As long as the stopping criterion is not satisfied, the population will evolve during a certain number of iterations [59]. This method is applied in our research work to develop solutions for the joint scheduling problem. Therefore, an entire section in this chapter is focused on this specific method to describe its principle and the different stages of the algorithm.

4.4.3 Multi-agent systems

Another popular methodology to tackle the joint scheduling issue for fleet of systems is to use a Multi-Agent System (MAS). It is an approach to model and develop applications in which decisions are decentralized by nature within agents [24]. It is part of what is known as Distributed Artificial Intelligence. The autonomous agents interact with each other directly or indirectly through a shared space, called environment, to achieve their individual and collective objectives. The multi-agent approach has shown its interest in conflict situations since it can be solved by negotiation techniques, in which the compromises moderate the satisfaction and frustration of agents.

Bencheikh et al. [24] use a multi-agent system to schedule simultaneously the production activities and maintenance operations for a set of machines. It consists in scheduling the production activities and conditioned maintenance operations according to the machines health states. They consider machines of limited availability that can be maintained if they become unable to perform a required function. In the case of an unavailable function, the machine can perform other operations that only need its available functions. The production tasks and the maintenance activities are synchronously planned according to the current and future health states of the machines defined by health assessment functions. The multi-agent model is inspired by the model Supervisor, Customers, Environment, Producers (SCEP). They add the maintenance component to this existing model. The multi-agent system environment is a blackboard on which the customers, producers and maintainers broadcast their demands. The cooperation between customer agents, producer agents and maintenance agents is performed synchronously through the background environment agent. The supervisor is the agent who controls the negotiation between the different agents and gives them access to the blackboard to dispatch their requests. Each customer agent manages the construction of one order given

by the customers. The customer agent associates to each operation composing the article an object in the environment. The main objective of each customer agent is to respect the due dates claimed by its corresponding customer. Each producer agent manages one machine. It schedules the operations the machine is able to perform according to the machine health status and makes the requests for the needed maintenance operations. Each maintenance agent manages one maintainer with specific skills of one service. A similar methodology has been used by Coudert et al. [44] with a mix of fuzzy logic to obtain a cooperative scheduling for production and maintenance. Feng et al.[58] apply the multi-agent system methodology to obtain the optimal configuration of maintenance personnel and the maintenance personnel dispatching scheme to ensure that missions can be performed with reliable aircrafts.

4.4.4 Game theory

Sometimes, decisions need to be made to ensure that a fleet of systems has the right reliability level to complete a mission. This problem is related to the selective maintenance topic. In this case, some methodology has been developed based on the game theory.

Feng et al. [56] study the fleet condition-based maintenance (CBM) planning problem for aircrafts. The objective is to define the set of aircrafts that will be dispatched on a mission while ensuring a fixed reliability level. The mission requirements, resource constraints and aircraft statuses are considered to find an optimal strategy set. The fleet CBM problem is treated as a two-stage dynamic decision-making problem. The aircrafts are divided into dispatch and standby sets. A heuristic hybrid game (HHG) approach, composed of a competition game and a cooperative game, is proposed on the basis of a heuristic rule. In the dispatch set, a competition game approach is proposed to search for a local optimal strategy matrix. A cooperative game method for the two sets is also proposed to ensure global optimization. Yang et al. [186] study the selective maintenance problem to identify and perform maintenance actions necessary for fleet mission success. They consider a fleet of aircrafts that is required to perform phase missions with short scheduled breaks. They aim at reducing the repair frequency and cost. The constraint is the reliability of the phased mission and the variables are the remaining useful lifetimes (RUL) of all the key subsystems. The equipment can be clustered into three stages based on the health status before each wave of missions and a heuristic game framework with state backtracking is proposed for the three stages to solve the problem. The second stage algorithm is used to select the dispatched equipment for the current wave and minimize maintenance while the third stage algorithm ensures that sufficient equipment is available for the next mission wave by performing necessary maintenance operations.

4.5 Genetic Algorithm

Lots of large-scale combinatorial optimization problems have been defined in the literature. Many of them have been proven NP-hard like the joint scheduling problem for production

and maintenance. For small spaces, classical exhaustive methods are sufficient to solve the problem. But for larger spaces, special artificial intelligence techniques must be applied. Genetic algorithms are among such techniques: they are stochastic algorithms whose search methods model some natural phenomena like genetic inheritance and Darwinian strife for survival [113].

To jointly schedule maintenance operations and missions for a vehicle or a fleet of vehicles, we chose to use genetic algorithms. From what we described in section 4.2, this method is often used in the literature to solve this class of problems. Genetic algorithm is part of the evolutionary algorithms that enable to find high-quality solutions, even if they may be sub-optimal, to optimization and search problems.

In this section, we define what a genetic algorithm is, the components used to develop one as well as its way of working. Based on these elements, we identify the advantages and drawbacks offered by such a resolution method. This description is mainly based on the information found in the books of Michalewicz [113] and of Sivanandam and Deepa [156].

4.5.1 Definition

As written before, genetic algorithms are part of a class of algorithms called evolutionary algorithms. They represent stochastic optimization methods based on the concepts of natural selection and genetics. It is a meta-heuristic method that enables to obtain a sub-optimal solution for a problem in a reasonable computation time. It is very useful for combinatorial problems that cannot be solved with an exact method or that need a too long computation time to reach a solution.

A genetic algorithm performs a multi-directional search by maintaining a population of potential solutions and encourages information formation and exchange between these directions [113]. The population, composed of potential solutions, undergoes a simulated evolution. At each generation, the relatively fit solutions reproduce, while the relatively bad solutions die. To make the difference between the fit and bad solutions, an objective function is used to evaluate the individuals.

A genetic algorithm is defined with five components:

- a genetic representation for potential solutions of the problem,
- a way to create an initial population of potential solutions,
- an evaluation function rating solutions in terms of their goodness of fit,
- genetic operators that alter the composition of children,
- values for various parameters used by the genetic algorithm.

The objective of the genetic algorithm is to find the fittest individual, i.e. the best solution for the problem. It makes a population of individuals evolve through an iterative process by

applying genetic operators like selection, crossover and mutations. These operations alter the individuals but the fittest ones are kept from one iteration to another until the algorithm convergence.

4.5.2 Principle

A genetic algorithm (GA) handles a population of possible solutions. Each solution is represented through a chromosome, itself composed of a set of genes. The solution can be encoded through a binary representation, the traditional one, or with float numbers. A set of reproduction operators also have to be determined. These operators are applied directly on the chromosomes and are used to perform mutations and recombinations over solutions of the studied problem. Defining the appropriate representation and operators is essential because the GA behaviour depends a lot on it. Frequently, it can be really difficult to find the right representation that respects the search space structure, and coherent and relevant reproduction operators according to the problem properties.

The selection stage is supposed to be able to compare each individual in the population. It is done with the help of a fitness function, i.e. an objective function that enables to evaluate how good a candidate solution is. Each chromosome has then an associated value corresponding to its fitness. The optimal solution is the one maximizing the fitness function. Naturally, if the problem aims at minimizing a cost function, the adaptation is quite easy. Either the cost function is transformed into a fitness function or the selection is adapted so that they consider the individuals having a low evaluation function as better individuals.

Once the reproduction process and the fitness function have been properly defined, a GA evolves based on a basic structure. It starts by generating an initial population of individuals. The first population must offer diversity among the individuals to avoid an early convergence of the algorithm towards a local optimum. The gene pool has to be as large as possible so that any solution of the search space can be generated. Most of the time, the initial population is randomly generated. But, sometimes, some heuristic rules can be applied to define already good solutions that have the researched properties to guide the algorithm in the right direction and accelerate the convergence.

Then, the genetic algorithm loops over an iterative process to make the population evolve. Each iteration consists of the following stages:

- **Selection:** The first stage consists in selecting the individuals for reproduction. Different operators exist but they all have the same objective: selecting most often the best individuals with respect to their fitness values than the poor ones.
- **Reproduction:** In the second stage, offspring is bred by the selected individuals. For generating new individuals, crossovers (also called recombination) and mutations are applied.
- **Evaluation:** The fitness of the new individual is then evaluated.

- **Replacement:** During the last step, individuals from the old population are killed and replaced by the new ones.

The general principle can then be summed up [156]:

- **[Start]** Generate random population composed of suitable solutions for the problem.
- **[Fitness]** Evaluate the fitness of each individual in the population.
- **[New population]** Create a new population by repeating the next steps until the new population is complete:
 - **[Selection]** Select two parents chromosomes according to their fitness (the better fitness, the bigger chance to get selected).
 - **[Crossover]** Cross over the parents to obtain children with a crossover probability. If no crossover is performed, children are the exact copies of their parents.
 - **[Mutation]** Mutate a new child at each position in the chromosome with a mutation probability.
 - **[Accepting]** Place new children in the population.
- **[Replace]** Use the new generated population for a further sum of the algorithm.
- **[Test]** If the end condition is satisfied, the algorithm stops and returns the best solution in the current population.
- **[Loop]** Go back to the fitness evaluation stage (stage 2).

Reproduction or crossover is the process by which the genetic material in two or more parents is combined to obtain one or more children. In the fitness evaluation step, the individual's quality is assessed. Mutation is performed on one individual to produce a new version of it, where some of the original genetic material has been randomly changed. The selection process helps to decide which individuals are to be used for reproduction and mutation in order to produce new search points.

There are three parameters to tune in order to improve the performance and optimize the convergence of the algorithm: the population size, the probability of crossover and the probability of mutation. The end condition can also be based on a parameter to tune if the stopping criterion is defined by a number of iterations.

There are many other methods used in the literature to solve optimization problems, but the genetic algorithms differ from conventional optimization techniques. They operate with coded versions of the problem parameters rather than parameters themselves. It means that they work with the coding of the solution set and not the solution itself. Most of the conventional optimization techniques search for a single solution while GA uses population of solutions. It improves the chance of reaching a global optimum and helps avoiding local stationary optimum. The algorithm uses a fitness function for evaluation rather than derivatives.

Hence, it can be applied to any kind of continuous or discrete optimization problem. Finally, genetic algorithms use probabilistic transition operators while conventional methods apply deterministic ones for continuous optimization.

4.5.3 Advantages and drawbacks

Sivanandam and Deepa [156] enumerate the main advantages and the main drawbacks of genetic algorithms. The advantages include:

- parallelism;
- the solution space is wider;
- the fitness landscape is complex;
- it is easy to find a global optimum;
- we can consider multi-objective function;
- they are easily modified for different problems;
- they can handle large search spaces;
- they are good for multi-modal problems as they return a set of solutions;
- they are robust to difficulties in the evaluation of the objective function;
- they are resistant when it comes to be trapped in local optima;
- they perform well for large-scale optimization problems;
- they can be used in a huge variety of optimization problems.

However, they still have their limitations. The major difficulties lie in the following points:

- identifying the fitness function;
- defining the representation of individuals for the problem;
- occurrence of premature convergence;
- they have trouble finding the exact global optimum;
- they are not good at identifying local optima;
- they need to be coupled with local search technique;
- there is no effective termination;

- they require a large number of fitness function evaluations (for each individual at every iteration);
- their configuration is not straightforward as we have to choose various parameters, like the population size, the mutation and crossover rate, the selection method, that have a significant impact on their performance.

4.6 Conclusion

This chapter offers a review of the existing research topics when it comes to jointly schedule maintenance and production for either a single system or a fleet of systems. It shows that there is a huge variety of problem descriptions that consider different assumptions to solve them. Many of the works are focused on dealing with the joint scheduling problem for a single system in a static framework and do not integrate any monitoring information to adapt the defined schedule. However, some publications dig into the dynamic joint scheduling problem. The hardest question to answer is to estimate the impact of rescheduling in terms of costs and logistics. Finally, some researchers start to study the joint scheduling problem for a fleet of systems. In addition to the variety of joint scheduling problem descriptions, there is also the diversity of optimization methods for solving them. The choice among one of them depends a lot on the problem size, the systems to study and the objectives to reach.

However, the existing approaches do not enable to address the joint scheduling problem for maintenance and missions properly when we consider that the vehicles stochastically deteriorate over time. This uncertainty in the deterioration process has not yet been integrating in the joint scheduling problem for a fleet of vehicles.

In the following chapters, we choose to use a meta-heuristic method, the genetic algorithm, to solve the joint scheduling problem for a single vehicle and a fleet of vehicles in both static and dynamic frameworks. This choice has mainly been motivated by the fact that, as the joint scheduling problem is a NP-hard combinatorial problem, an exact method is very quickly overtaken in terms of computation time, even for the static case with a single vehicle. Another method is then necessary to obtain a satisfying solution in a reasonable computation time. Moreover, as we try to solve the joint scheduling problem step by step, by considering the easiest case first and adding complexity, the genetic algorithm is quite practical. Indeed, it is easily modified to fit to the new study case. That is why, we focus on the description of this methodology in this chapter.

In this context, we propose to develop, in the following chapter, a genetic algorithm-based methodology to solve the joint scheduling problem for missions and maintenance operations for a single vehicle in the static case.

Joint scheduling of missions and maintenance for a vehicle: the static case

Contents

5.1	Problem definition	94
5.1.1	Hypotheses and constraints	94
5.1.2	Objective	95
5.2	Resolution approach description	97
5.2.1	Vehicle deterioration model	97
5.2.1.1	Health deterioration model	97
5.2.1.2	Missions impact on the deterioration	98
5.2.2	Decision criteria	103
5.2.3	Different resolution methods	106
5.2.3.1	An exact resolution method	106
5.2.3.2	A genetic algorithm based method	108
5.3	Performance of the method	115
5.3.1	Application example	115
5.3.2	Performance analysis	118
5.3.3	Sensitivity study	121
5.3.3.1	Impact of the ratio between the preventive and corrective maintenance costs	121
5.3.3.2	Effect of the variation of the block filling condition	123
5.3.3.3	Influence of the missions variance changes	124
5.3.4	Limitations of the genetic algorithm	128
5.4	Conclusion	128

The previous chapter presents the existing methods to jointly schedule missions and maintenance operations for a system. In most cases, one of the two activities is favoured when it comes to define a schedule. However, very few methods have been proposed to schedule both activities at the same time. The interest in simultaneously scheduling missions and maintenance operations is that each activity is considered as meaningful as the other. The

maintenance operations are then planned as closed as possible to the actual usage of the vehicle.

This chapter proposes a method to jointly schedule both the missions and the maintenance operations for a single truck. To start this problem, we place ourselves in a static case. It means that all the missions are available before the schedule definition and the final schedule cannot be modified no matter what event can occur during its completion.

In a first place, this chapter raises the question of how to schedule missions and maintenance at the same time and defines the different considered hypotheses to tackle this issue. The static scheduling method is then presented [140] and evaluated through application examples that draw a comparison with an exact method to analyse the performances of the proposed method. A sensitivity study is also led to evaluate the impact of some parameters introduced through the method.

5.1 Problem definition

This section introduces the problem of scheduling both missions and maintenance operations and describes the different elements necessary to its resolution.

5.1.1 Hypotheses and constraints

We consider a single truck that has a set of missions to complete. The truck health state deteriorates over time according to its activity. Its activity is diverse and depends on the missions operating conditions. Indeed, the vehicle operates in missions with different usage severities, characterized by durations but also by different environment parameters, such as road condition or topography. The truck usage has then to be modelled to integrate the missions severity levels.

A vehicle deterioration model is then necessary to be able to make the best decisions regarding the maintenance and mission scheduling. This model is essential to define a maintenance model adapted to the vehicle usage and to schedule at best the missions without having a too high failure risk during the missions completion.

We consider that the cost associated with a preventive maintenance operation is C_0 . If there is a failure during a mission completion, the maintenance operation corresponds to a corrective maintenance operation and the charged maintenance cost is then C_f . We also consider that every maintenance operation is perfect. It means that every time a maintenance operation is performed, the vehicle deterioration state is always brought back to an « As Good As New » state.

5.1.2 Objective

The objective aimed with this problematic is to jointly optimize the vehicle predictive maintenance and its mission planning using the deterioration information described by the missions features and the vehicle deterioration model. [Figure 5.1](#) describes the framework to schedule three missions M_1 , M_2 and M_3 and the maintenance operation MO according to the health state. This figure aims at globally depicting the process to build a schedule. However, as no real-time deterioration measurements are available, the decision-making process is based on estimated failure probabilities. This process will be detailed in [Section 5.2.1](#).

The preventive maintenance operations have to be scheduled to prevent immobilizing failures, to maximize the vehicle availability and not to disturb the missions progression. To fill these conditions, the final schedule is defined as a series of mission blocks interrupted by maintenance operations to restore the vehicle deterioration back to 0. In [Figure 5.1](#), the blocks would be $[M_3, M_1]$ and $[M_2]$ separated by a maintenance operation MO .

The schedule optimization is based on the total maintenance cost. This cost is composed of two parts: the first one corresponds to the costs associated with preventive maintenance while the second one includes the corrective maintenance costs associated with the missions failure risk (see [Section 5.2.2](#)). The maintenance and mission model optimization enables to define the best moments to end missions blocks and schedule maintenance operations as well as to fill the different blocks in the best possible way to minimize the number of scheduled maintenance operations.

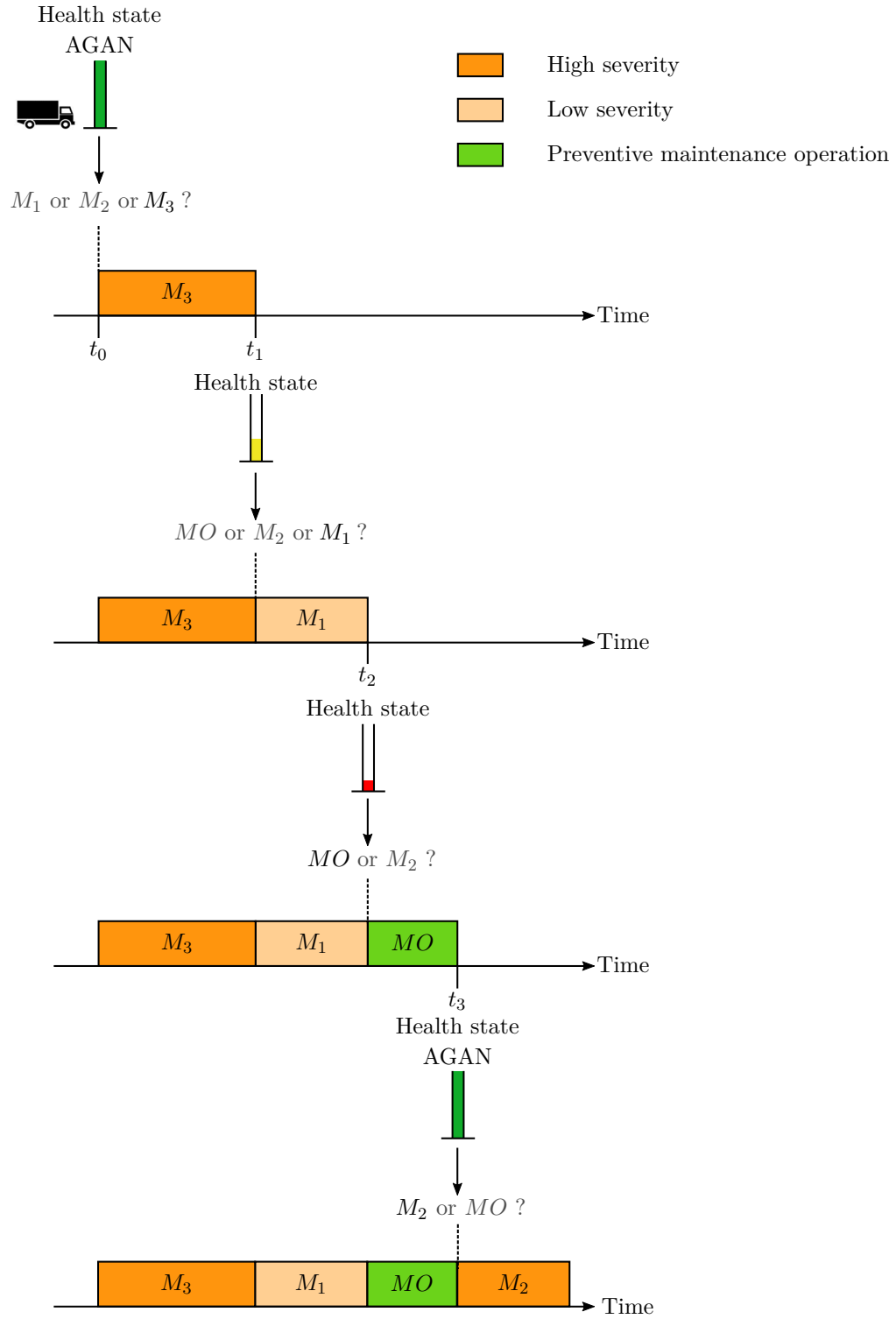


Figure 5.1: Framework to schedule the missions M_1 , M_2 and M_3 and the maintenance operations MO according to the health state

5.2 Resolution approach description

This section describes the method applied to define the schedule for missions and maintenance operations. It is composed of three subsections. The first one depicts the vehicle deterioration model to take into account the missions impact on the deterioration due to changes regarding the vehicle operating conditions. The second part defines the decision criterion on which the decision-making process is based. Finally, the third part describes the genetic algorithm developed to jointly optimize the missions and the maintenance operations scheduling.

5.2.1 Vehicle deterioration model

5.2.1.1 Health deterioration model

The vehicle for which we want to schedule the activity is considered as a single-component system. It is then characterized by a global health indicator.

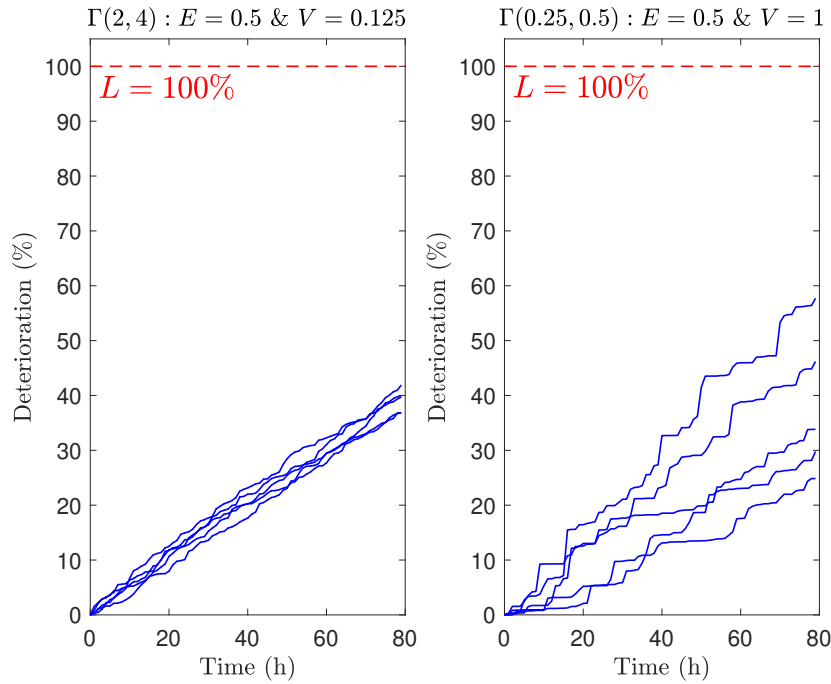


Figure 5.2: Deterioration trajectories following two Gamma processes with the same expected value E but with different variance V

We consider the deterioration as a stochastic phenomenon rather than a deterministic one. As explained in [Section 2.4.1](#), there are two classes of stochastic deterioration models: the discrete deterioration models and the continuous ones. The continuous deterioration models are more suitable to represent observable phenomena on industrial systems [\[97\]](#) as most components are mechanical and subject to gradual wear. It is then relevant to model

the deterioration of the vehicle by a continuous deterioration model.

We choose to use a Gamma process (Section 2.4.2.2) to model the deterioration evolution [168]. This process is well suited because, as its increments are always increasing, the vehicle health state can only improve thanks to an external intervention. In our case, this intervention is a maintenance operation.

When the vehicle cumulated deterioration exceeds the failure threshold L , a failure occurs. To make decisions whether to deploy the vehicle on a mission or to send it to the workshop for a maintenance operation, we use the failure probability estimation at the end of the considered mission. However, the deterioration level changes according to the missions the vehicle has already completed and the maintenance operations already performed.

Figure 5.2 represents the deterioration trajectories generated with two Gamma processes whose respective pairs of shape and scale parameters are $(\alpha = 2, \beta = 4)$ and $(\alpha = 0.25, \beta = 0.5)$. When the variance increases, the deterioration trajectories are more spread. It then increases the uncertainty of the deterioration evolution when using a Gamma process with such parameters.

5.2.1.2 Missions impact on the deterioration

- **Link mission and deterioration**

The missions correspond to the deliveries the vehicle has to complete on a fixed time horizon. They are characterized by different durations, according to the distance the vehicle has to cover, and different severities, that affect the deterioration evolution differently.

Indeed, the vehicle evolves in a dynamic environment that influences its deterioration. The environment variations come from changes in the missions features during the vehicle lifetime. The deterioration-threshold failure model allows one to integrate the missions features through the modification of the deterioration parameters. It is assumed that these changes have a time-related impact on the vehicle deterioration process, modelled by a change in the deterioration speed.

The Gamma process, used to model the vehicle deterioration evolution, is then a process with varying parameters. Each mission m is associated with a pair of parameters α_m and β_m corresponding to the shape and scale parameters of the Gamma process modelling the vehicle deterioration evolution.

However, a mission to mission deterioration model is difficult to use as the missions will be grouped into different blocks separated by maintenance operations in the final schedule. If we do not have a deterioration measurement at the end of each mission, we cannot just estimate the remaining useful life using the parameters of the potential next mission and the current deterioration state. The operational environment can change from mission to mission inside the same block. In this context, as maintenance operations are directly related to the vehicle health state at the end of the blocks,

modelling the deterioration evolution at the block level, and not at the mission level, is necessary.

- **Approximated deterioration process on a block**

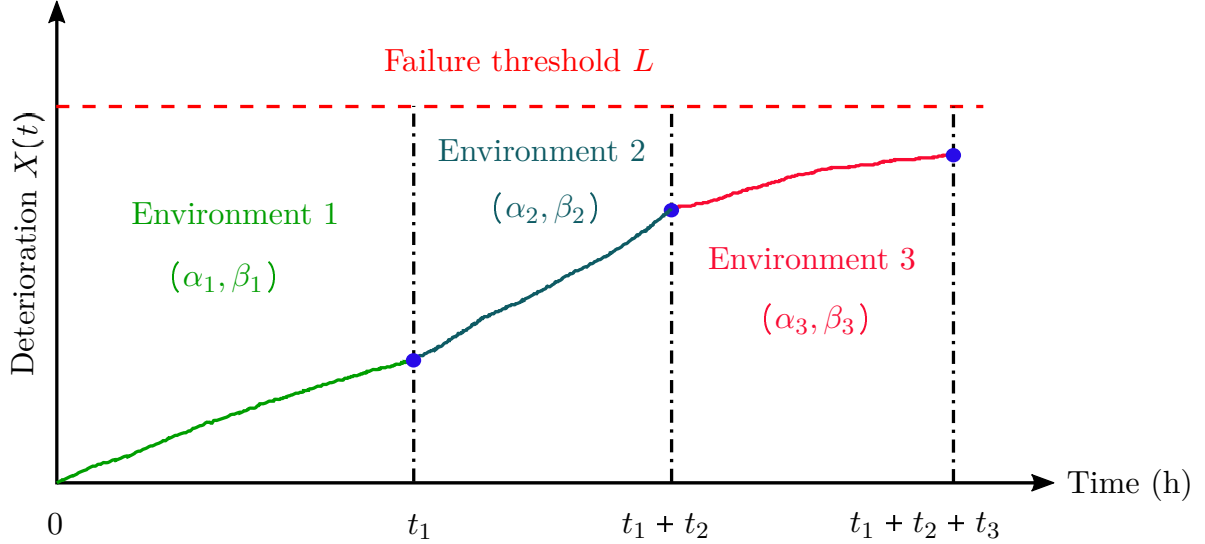


Figure 5.3: Deterioration evolution in a block with 3 missions

When a block, defined as a group of missions, is composed of p different missions, it means that $p - 1$ environment changes occur inside the same block. These changes represent the different missions severity levels. Figure 5.3 shows an example of a block composed of three different missions M_1 , M_2 and M_3 . We then have three different operational environments and the only known information is that the deterioration level at the beginning of the block is equal to 0. No other information is recorded during the completion of the block by the vehicle.

The cumulated deterioration between $t_0 = 0$ and $t_1 + t_2 + t_3$ corresponds to the sum of the deterioration increments between $t_0 = 0$ and t_1 , t_1 and $t_1 + t_2$ and between $t_1 + t_2$ and $t_1 + t_2 + t_3$. Let us respectively denote f_1 , f_2 and f_3 the increments probability densities for the missions M_1 , M_2 and M_3 . f_1 , f_2 and f_3 are Gamma laws whose parameters are respectively (α_1, β_1) , (α_2, β_2) and (α_3, β_3) .

Let us also define $D_1 = t_1$ and $D_2 = t_1 + t_2$ the dates when an environment change occurs. Khoury [80] explains that the density for the deterioration increment of the block, denoted f , is given by the convolution product of the increments densities of the missions inside the block (Eq. 5.1). But the more numerous the environment changes are, the more complex the reliability expression is to compute.

$$f(x) = (f_1 * f_2 * f_3)(x) = \int_0^x \int_0^{x-u_1} \int_0^{x-u_1-u_2} f_3(u_3, \alpha_3(t_1 + t_2 + t_3 - D_2), \beta_3) \quad (5.1)$$

$$f_2(\alpha_2(D_2 - D_1), \beta_2)$$

$$f_1(\alpha_1(D_1 - t_0), \beta_1)$$

$$du_3 du_2 du_1$$

To avoid the computation of this convolution integral, it would then be relevant to model the vehicle deterioration evolution in the block by an equivalent Gamma process $Ga(\alpha_e, \beta_e)$, based on the Gamma process properties and the environment changes knowledge [97].

If the scale parameters β are all equal, then we have the exact expression because we consider Gamma laws. Otherwise, we need to use an approximation as proposed in the article [134].

For this approximation, the idea is to estimate the average value and the variance of the equivalent Gamma process. They are respectively approximated weighted mean values of the average values and of the variances related to the Gamma processes describing the deterioration evolution for each mission independently (Eq. 5.2). The different weights are the proportions of the block duration spent in each environment. The equivalent Gamma process average value and variance for the block described in Figure 5.3 can then be defined as follows:

$$\begin{aligned} \frac{\alpha_e}{\beta_e} &\approx \frac{t_1}{t_1 + t_2 + t_3} \cdot \frac{\alpha_1}{\beta_1} + \frac{t_2}{t_1 + t_2 + t_3} \cdot \frac{\alpha_2}{\beta_2} + \frac{t_3}{t_1 + t_2 + t_3} \cdot \frac{\alpha_3}{\beta_3} \\ \frac{\alpha_e}{\beta_e^2} &\approx \frac{t_1}{t_1 + t_2 + t_3} \cdot \frac{\alpha_1}{\beta_1^2} + \frac{t_2}{t_1 + t_2 + t_3} \cdot \frac{\alpha_2}{\beta_2^2} + \frac{t_3}{t_1 + t_2 + t_3} \cdot \frac{\alpha_3}{\beta_3^2} \end{aligned} \quad (5.2)$$

More generally, if we have p missions composing a block, the average value and the variance for the equivalent Gamma process followed by the vehicle deterioration level can be defined as in Eq. 5.3. Note that if all the Gamma processes related to the missions have the same scale parameter β and the same duration, the result given by the convolution product is exactly the Gamma process $Ga(\sum_{i=1}^p \alpha_i, \beta)$.

$$\frac{\alpha_e}{\beta_e} = \frac{1}{T} \sum_{i=1}^p t_i \frac{\alpha_i}{\beta_i} \quad \text{and} \quad \frac{\alpha_e}{\beta_e^2} = \frac{1}{T} \sum_{i=1}^p t_i \frac{\alpha_i}{\beta_i^2} \quad (5.3)$$

To evaluate the quality of the results given by this equivalent Gamma process approximation, we propose a study case through a simulation framework.

- **Validation of the equivalent distribution**

Let X_1 and X_2 define two independent and identically distributed random variables respectively following the Gamma distributions $\Gamma(a_1.T_1, \beta_1)$ and $\Gamma(a_2.T_2, \beta_2)$. These two variables represent the deterioration value at the end of a certain time for two different operational environments. The vehicle respectively spends T_1 and T_2 time units in each operational environment and the shape parameters of the distributions are respectively $\alpha_1 = a_1.T_1$ and $\alpha_2 = a_2.T_2$. The values of the different parameters are detailed in Table 5.1.

We generate $N = 100000$ realizations for each random variable. Let $x_1 = (x_{11}, x_{12}, \dots, x_{1N})$ and $x_2 = (x_{21}, x_{22}, \dots, x_{2N})$ respectively be realizations for the random variables X_1 and X_2 .

Table 5.1: Parameters values

Parameters	Values
a_1	1
β_1	0.2
T_1	5
a_2	0.3
β_2	0.01
T_2	10

Then, we sum each realization of X_1 with one of X_2 to obtain realizations corresponding to the sum of the random variables X_s . We obtain the realizations x_s for the random variable X_s (Eq. 5.4).

$$x_s = \begin{pmatrix} x_{11} + x_{21} \\ x_{12} + x_{22} \\ \vdots \\ x_{1N} + x_{2N} \end{pmatrix} \quad (5.4)$$

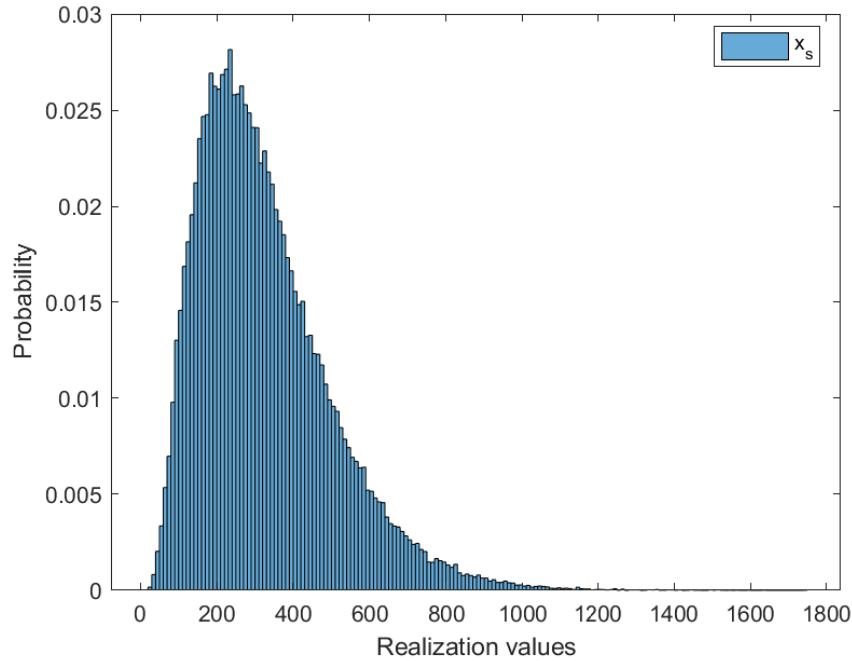


Figure 5.4: Histogram (empirical probability distribution function) of the realizations of x_s , corresponding to the sum of the realizations of x_1 and x_2 for each Gamma distribution

Thanks to x_s , the distribution followed by X_s can be plotted (Figure 5.4). As X_s is the sum of random variables following a Gamma distribution, X_s also follows a Gamma

distribution [97]. The shape and scale parameters defining this distribution can be estimated. The « Distribution Fitting Tool » (DFT) available in the optimization library from Matlab enables to estimate respectively the shape parameter at 3.6284 with a standard error equal to 0.0155 and the scale parameter at 0.0112 with a standard error equal to 0.4116.

In parallel, the parameters corresponding to the equivalent distribution are computed based on Eq. 5.2 and $N = 100000$ realizations of x_1 , x_2 and x_s are generated. The obtained distribution as well as the distribution parameters estimation can then be compared to the previous ones.

Based on Eq. 5.2, the values for the estimated shape and scale parameters can be obtained thanks to the equations 5.5 and 5.6. They are respectively equal to 3.5062 and 0.0108.

$$\alpha_e = \frac{1}{T} \frac{\left(\sum_{i=1}^p t_i \frac{\alpha_i}{\beta_i} \right)^2}{\sum_{i=1}^p t_i \frac{\alpha_i}{\beta_i^2}} \quad (5.5)$$

$$\beta_e = \frac{\sum_{i=1}^p t_i \frac{\alpha_i}{\beta_i}}{\sum_{i=1}^p t_i \frac{\alpha_i}{\beta_i^2}} \quad (5.6)$$

Figure 5.5 displays a comparison between the equivalent distribution and the Gamma distribution followed by X_s as the sum of two random variables X_1 and X_2 . The two histograms have the same shape and are quite similar. We can compute the relative errors for the shape and scale parameters to estimate the quality of the method to compute the equivalent distribution. The relative error for the shape and scale parameters are respectively equal to 3.4% and 3.6% (Eq. 5.7).

$$\begin{aligned} \varepsilon_{shape} &= \frac{3.5062 - 3.6284}{3.6284} = -0.0337 \\ \varepsilon_{scale} &= \frac{0.0108 - 0.0112}{0.0112} = -0.0357 \end{aligned} \quad (5.7)$$

Based on the study case results, the method to estimate the equivalent distribution is considered as a good approximation. It is a huge advantage to use such a method rather than using the convolution product to compute the distribution. Indeed, it will enable to reduce the computation time when defining the complete missions/ maintenance schedule for the vehicle.

The deterioration model for each block of missions based on the missions features is a key point to estimate the failure probability of a block and make decision to define the joint schedule. The next step is to integrate this model when defining the decision criterion.

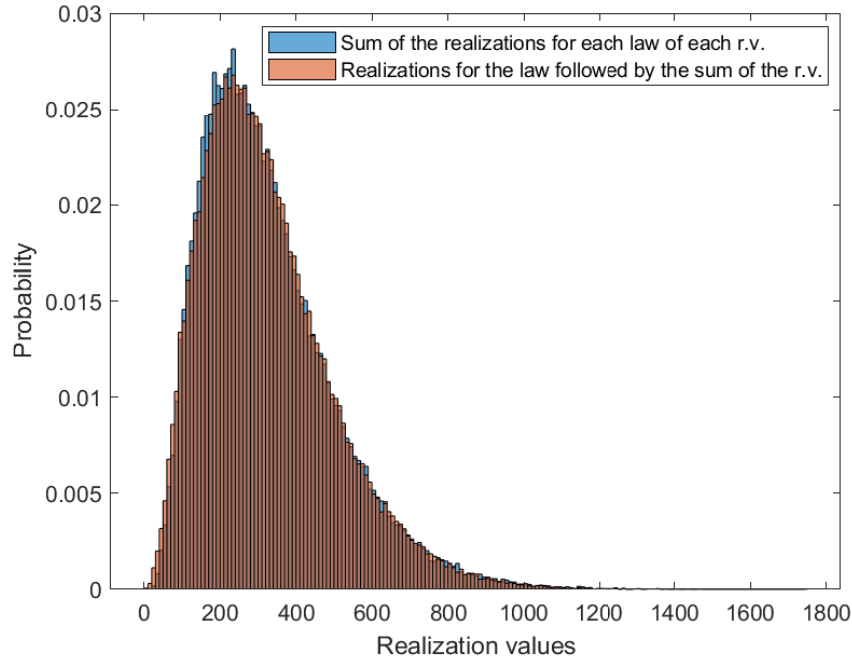


Figure 5.5: Comparison between the real distribution of X_s , based on the realizations x_s , and the equivalent distribution (r.v. stands for random variable)

5.2.2 Decision criteria

The schedule we would like to obtain is composed of mission blocks separated by preventive maintenance operations. An optimization criterion could then be defined based on the maintenance cost associated with the schedule. Such a criterion relies on two elements:

- The maintenance cost for all the scheduled preventive maintenance operations. Each preventive maintenance operation costs C_0 and occurs at the end of each mission block composing the schedule.
- The corrective maintenance cost related to the failure occurring during the different mission blocks. Each corrective operation costs C_f .

To define an optimal schedule for the missions and the maintenance operations, a balance has to be found between preventive and corrective maintenance, i.e. in our setting between the number of blocks and the blocks filling. If the schedule is defined with few blocks composed of many missions, more corrective maintenance operations will be performed than preventive maintenance operations. On the contrary, if we have lots of blocks but with few missions, we will perform more preventive maintenance operations than corrective maintenance operations.

The block filling is directly related to the deterioration evolution of the vehicle, denoted $X(t)$, $t > 0$. To determine the possible number of missions in the different blocks, the equiv-

alent deterioration process for each block is used to estimate the failure probability on each block. If we consider the block described in Figure 5.3, the failure probability of this block (Eq. 5.8) is based on the equivalent Gamma process $Ga(\alpha_e, \beta_e)$. It corresponds to the probability that the cumulated deterioration at the end of the block duration T exceeds the failure threshold L , knowing that $X(t_0)$, the deterioration at $t_0 = 0$, is equal to 0 [81, 97].

$$P(X(T) - X(t_0) \geq L) = \frac{\Gamma(\alpha_e(T - t_0), L\beta_e)}{\Gamma(\alpha_e(T - t_0))} \quad (5.8)$$

where $\Gamma(\cdot)$ is the Gamma function and $\Gamma(\alpha, L\beta)$ is the upper incomplete Gamma function defined by Eq. 5.9.

$$\forall t > 0, \quad \Gamma(\alpha t, L\beta) = \int_{L\beta}^{\infty} u^{\alpha t - 1} e^{-u} du \quad (5.9)$$

We define two decision criteria to estimate the maintenance cost for a joint schedule π composed of N_b blocks of missions. The criteria are composed of two parts: the first part corresponds to the preventive maintenance costs related to the preventive maintenance operations occurring at the end of each mission block. The second part estimates the costs associated with corrective maintenance operations.

For the first decision criterion C_1 (Eq. 5.10), this second part is estimated based on the probability to have one failure in each mission block. The probability to have a failure in the block k is denoted $\mathbb{P}_f(k)$ and is computed as explained in Eq. 5.8.

- First criterion (single failure criterion)

$$C_1(\pi) = \sum_{k=1}^{N_b} (C_0 + C_f \mathbb{P}_f(k)) \quad (5.10)$$

- Second criterion (multiple failures criterion)

$$C_2(\pi) = \sum_{k=1}^{N_b} \left(C_0 + C_f \sum_{i=1}^{N_m(k)} \mathbb{P}_{f_k}(D \geq iL) \right) \quad (5.11)$$

where $N_m(k)$ is the number of missions in the block k , D the deterioration level and $\mathbb{P}_{f_k}(D \geq iL)$ the probability to exceed the threshold L for the i^{th} time in the block k .

For both criteria, a parameter \mathbb{P}_{max} is defined to condition the block filling. This parameter acts on the maximum admissible failure probability for a block and its value is between 0 and 1. Adding such a parameter enables to reduce the number of schedule possibilities as it is more likely to dispatch a vehicle on a mission block whose failure probability is low to avoid at best corrective maintenance operations.

The second criterion is an adaptation of the first one to integrate the possibility to have more than one failure in the different mission blocks. We denote this second criterion C_2 which considers that in a block composed of m missions, no more than m failures can occur. Considering multi failures in a block leads to estimate the expected failure occurrences in the considered block.

The replacement process is such that once the deterioration level oversteps the failure threshold L , a corrective maintenance operation is performed. The vehicle deterioration level is then brought back to 0 as we consider that the maintenance operations are perfect. Based on the equivalent Gamma process, we can characterize the vehicle deterioration evolution when completing the block. The probability to have two failure occurrences for instance in the block defined by Figure 5.3 can then be estimated by $P(X(T) - X(t_0) \geq 2L)$ where T is the block duration and t_0 the date when the block starts. The principle is the same as with a replacement process but we do not reset the deterioration level at 0 after a failure (Figure 5.6). The criterion C_2 for the schedule π composed of N_b blocks can be defined by the following equation.

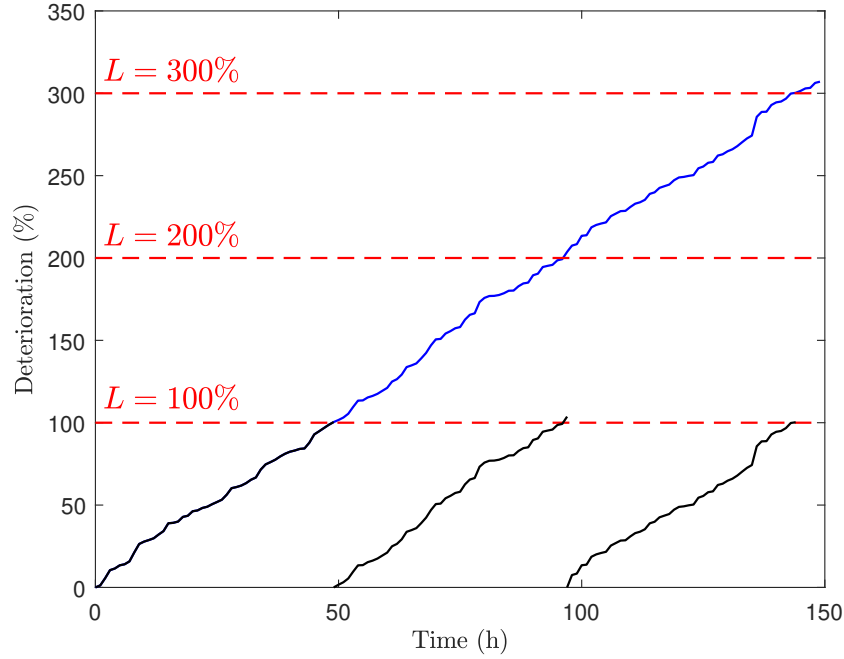


Figure 5.6: Principle of the replacement process: representation of $\mathbb{P}_{f_k}(D \geq iL)$

The two criteria C_1 and C_2 are used as the optimization criteria for the genetic algorithm that have been developed to solve the missions/maintenance joint scheduling problem.

5.2.3 Different resolution methods

After defining the different criteria that can be used to order the different missions and schedule the different maintenance operations, we are now interested in the method that can be applied to solve the joint scheduling problem. To solve this optimization problem, the major difficulties appear when the number of missions to schedule increases. When we try to define the most suitable blocks of missions, it is possible to group the missions by two, by three and so on. It leads to a very high number of possibilities. To avoid the combinatorial explosion and to reduce the computation time, some approximated resolution methods can be considered. They enable to drastically reduce the number of candidates that have to be considered.

To estimate how significant the combinatorial explosion problem can be, we propose an exact method that considers all the possible schedules respecting the block filling conditions characterized by \mathbb{P}_{max} and their associated criterion values to minimize the chosen criterion.

To avoid the combinatorial explosion and to reduce at best the computation time, a genetic algorithm (GA) is developed. The genetic algorithms (GAs) represent stochastic optimization methods based on the concepts of natural selection and genetics [75, 156]. They are approached resolution methods. We use this meta-heuristic method due to its adaptability regarding the definitions of the individuals and the operators as well as its performances to reach a good solution in a satisfying computation time.

5.2.3.1 An exact resolution method

An exact resolution has been firstly developed to start digging into the joint scheduling problem and see the amount of possible schedules that could be generated while respecting the block filling condition characterized by \mathbb{P}_{max} .

This method generates all the possible schedule definitions based on the block filling condition $P(block) \leq \mathbb{P}_{max}$ and the number of available missions. It is based on a recursive function that is defined as follows.

- Firstly, we generate all the possible blocks according to the available missions while respecting the block filling condition. To compute all the possible blocks that can be defined with n available missions, we need to define all the blocks composed of one, two, three, ..., n missions. For instance, defining the blocks composed of two missions means finding all the possible combinations of two missions among the n available ones.

Let us take an example with 4 missions to schedule. If we want to build blocks, we know that we can have blocks composed of either one, two, three or four missions (Table 5.2). The blocks composed of one mission correspond to schedule each mission in separated blocks. To obtain the blocks composed of two missions, we need to generate all the groups of two missions by selecting two missions out of the four available ones. The

number of blocks is equal to $\binom{4}{2} = 6$. For the blocks of three missions, we define all the groups of three missions by selecting three missions out of four. It gives us $\binom{4}{3} = 4$ blocks of three missions. And finally, the block composed of four missions is the block with all the available missions inside.

Table 5.2: Possible blocks for $n = 4$ missions

Blocks: 1 mission	Blocks: 2 missions	Blocks: 3 missions	Blocks: 4 missions
[1]	[1, 2]	[1, 2, 3]	[1, 2, 3, 4]
[2]	[1, 3]	[1, 2, 4]	
[3]	[1, 4]	[1, 3, 4]	
[4]	[2, 3]	[2, 3, 4]	
	[2, 4]		
	[3, 4]		

When listing the different blocks, it is not necessary to consider the permutations of missions inside the blocks. Indeed, as the optimization criteria only consider the failure probabilities, we only need to know the proportion of time spent in each mission with respect to the total block duration. The order of the missions does not matter.

Once we have all the blocks possibilities, we can remove the non-feasible blocks i.e. the blocks whose failure probability is higher than \mathbb{P}_{max} .

- Then, we start building the possible schedules. We assign the first available block to the schedule and update the remaining missions to schedule after this block. If there is no remaining mission, a possible schedule has been found. Otherwise, we call the function again to find the possible blocks to assign knowing the missions already scheduled.
- This recursive function enables to generate all the possible schedules respecting the block filling condition.

Afterwards, we just have to compute the value of the chosen criterion for each one of them to find the schedule(s) minimizing the criterion.

As long as the number of missions to schedule for the vehicle remains small, the exact method can be used because the computation time remains small i.e. about a few minutes. However, with larger size problem such as $n = 9$ missions to schedule with $\mathbb{P}_{max} = 0.95$, the exact method needs more than 11 days to obtain the optimal schedule. This statement shows the interest of using an approached method such as a genetic algorithm to find a good solution, even if it is not the optimal schedule, in a suitable computation time.

In such a problem, it is not possible to use any standard scheduling methods insofar as the failure probability changes for each block according to the missions we put in it. Adding new missions into a block changes its failure probability.

5.2.3.2 A genetic algorithm based method

The second method applied to solve the static joint scheduling problem for one vehicle is a genetic algorithm. Generally, the first stage to build a genetic algorithm is to build an initial population of individuals. Then, the next stage consists in evaluating and sorting the different individuals thanks to a fitness function, i.e. an objective function. Afterwards, as long as the stopping criterion is not satisfied, genetic operations such as crossover and mutation are applied on the population to obtain better individuals. Figure 5.8 describes the general operating principle of the genetic algorithm to solve the static joint scheduling problem.

Individual representation

The individuals correspond to the candidate schedules for missions and maintenance operations. In the GA, a solution is obtained by sequencing the mission set into different blocks of missions. In a block, the missions are assumed to be completed one after the other. At the end of each block, a preventive maintenance operation occurs. For instance, if we have $n = 6$ missions to schedule, a candidate schedule π can be as defined in Eq. 5.12. The schedule is composed of three blocks of missions, meaning that three maintenance operations are scheduled.

$$\pi = \{ [6, 2] [5, 3, 4] [1] \} \quad (5.12)$$

Initial population

The first stage of the GA consists in generating an initial population P_0 composed of N_{pop} individuals. In many genetic algorithms, the initial population is randomly generated. It is a good strategy when we have no clue about how the best individuals are going to look like. That is why, we decide to generate between 60% and 70% of random individuals. Each mission is then randomly put in a block while respecting the block filling condition.

However, in our case, we have an idea of the composition of the best individuals. It depends a lot on the ratio between preventive and corrective maintenance costs ratio $\frac{C_0}{C_f}$. When the ratio value is close to 1, the best schedules will be composed of few blocks with many missions inside them. On the contrary, we will have many blocks with few missions inside them if the ratio value is closer to 0. To ensure that these hints are considered, we use special techniques. Around 20% of the population is generated using the First Fit (FF) heuristic method [42]. This technique firstly consists in generating a random permutation of the missions to schedule. Then, it takes each mission in turn and places it into the first block of missions with the lowest index that can accommodate it. Accommodating here means that the selected mission can be placed in a block as long as the failure probability does not exceed the maximum admissible failure probability \mathbb{P}_{max} .

The last 20% of individuals are generated using two heuristic methods respectively called First Fit Decreasing (FFD) and Best Fit Decreasing (BFD) [42]. For these two methods, the

first step consists in sorting the missions in a decreasing order with respect to their failure probabilities. Then, for FFD, the principle is the same as for the FF method. For the BFD method, each mission is taken in turn and placed in the first feasible block of missions having the smallest residual capacity. In our case, the smallest residual capacity corresponds to the highest failure probability. It means that we try to fill at best the most full blocks. Each heuristic method generates one candidate schedule. Block permutations are then performed on these solutions to generate the other individuals. Note that the individuals generated by these techniques respect the block filling condition.

The percentage to generate each part of the initial population are arbitrarily chosen. The objective is only to generate both individuals with a lot of blocks and with few blocks. However, the composition of the population is important because it enables to orientate the genetic algorithm convergence and reduce the computation time. In addition, when the maximum admissible failure probability \mathbb{P}_{max} becomes higher, the heuristic methods encourage to fill the blocks. It is a good point because as the constraint on the block building is relaxed, there is a high chance that the best schedule is composed of few blocks.

Evaluation

The evaluation stage consists in applying a fitness function, also called an objective function, to evaluate the quality of each individual in the population. The fitness function is related to the maintenance cost based criteria defined in [Section 5.2.2](#). The two fitness functions for an individual π are defined in Eq. 5.13. The best schedule is the one maximizing the fitness function.

$$Fit_1(\pi) = \frac{1}{C_1(\pi)} \quad \text{and} \quad Fit_2(\pi) = \frac{1}{C_2(\pi)} \quad (5.13)$$

Selection operator

After the evaluation stage, the parent candidates have to be selected among the initial population for the crossover and mutation stages. A method based on the realizations of tournaments between two candidates is implemented to select the N_{pop} parents. It is called the 2-tournament selection operator [113]. To set up this method, N_{pop} tournaments are realized between two individuals randomly chosen among the initial population. The winner of each tournament, i.e. the candidate having the highest value for the fitness function, is selected as parent.

Crossover operator

Once the parents have been selected, $\frac{N_{pop}}{2}$ parent pairs are randomly formed to go through the crossover operation. The crossover probability for each pair of parents is P_{cross} . The crossover aims at combining the selected parents to generate a better child candidate. Its principle is inspired by the crossover operator developed by Rohlfshagen and Bullinaria [144]. Both parents blocks are listed and we randomly select blocks that are not overlapping to build the two offspring individuals. It is to ensure that no mission is duplicated in the schedule.

Then, for each offspring, we test if there are some remaining missions. If it is the case, they are added to either an existing block or a new one, while respecting the block filling condition.

Parent 1	1,2,9,8	10,3,6	4,7,5	Parent 2	5,10	2,9,1,3	6,7,8	4
----------	---------	--------	-------	----------	------	---------	-------	---

1) Concatenation

1,2,9,8	10,3,6	4,7,5	5,10	2,9,1,3	6,7,8	4
---------	--------	-------	------	---------	-------	---

Selected block

2) Assignment of the parents blocks

Child 1	10,3,6	Child 2	10,3,6
---------	--------	---------	--------

Possible remaining parents blocks

1,2,9,8	4,7,5	4
---------	-------	---

Selected block

Child 1	10,3,6	1,2,9,8	Child 2	10,3,6	1,2,9,8
---------	--------	---------	---------	--------	---------

Possible remaining parents blocks

4

Selected block

Child 1	10,3,6	1,2,9,8	4	Child 2	10,3,6	1,2,9,8	4
---------	--------	---------	---	---------	--------	---------	---

3) Remaining missions assignment

Child 1: 7, 5

Child 2: 5, 7

Child 1	10,3,6	1,2,9,8	4,7,5	4	Child 2	10,3,6	1,2,9,8	4,7,5
---------	--------	---------	-------	---	---------	--------	---------	-------

Figure 5.7: Example of a crossover operation between two individuals

The different stages for a crossover between two parents are:

- The blocks of both parents are concatenated;
- A block is randomly selected among the ones from the parents and copied to the offspring individuals. In parallel, a list of the assigned missions is started;
- According to the already scheduled missions, a list of the blocks that can be assigned

is computed;

- As long as no blocks from parents can be selected, i.e. they all contain at least one mission already assigned, parent 1 and parent 2 are respectively scanned from left to right to find the missions that are not yet assigned;
- If there are still missions available, they are randomly assigned to blocks while respecting the maximum admissible failure condition \mathbb{P}_{max} . The order of assignment for child 1 and child 2 is the same as the apparition order of the remaining missions respectively in parent 1 and parent 2.

An example to illustrate the crossover operation is presented in [Figure 5.7](#).

Mutation operator

After the crossover operation, every child individuals is sent to the mutation stage. The probability to mute an individual is fixed at P_{mut} . The mutation operator is based on the Swap mutation [113] and consists in exchanging two randomly selected missions from two different mission blocks while respecting the block filling condition. The mutation stage is defined to increase the dispersion between the different individuals and for the algorithm to avoid reaching a local optimum. At the end of this stage, we obtain a population P_1 composed of N_{pop} individuals.

Population dispersion

We periodically evaluate the total population dispersion. This stage is important because it enables to avoid converging towards a local optimal solution. The total population includes both the individuals in P_0 and P_1 . It is then based on $2N_{pop}$ individuals. As explained by Ladj et al. [87], diversification and intensification are two major issues when it comes to build effect search algorithms. Diversification refers to the capacity to explore different regions of the research space while intensification refers to the ability to generate new solutions based on the high fitted solutions. For intensification, we focus on the search space regions where the solution are the fittest. To have a suitable population for the algorithm to converge while avoiding local optima, a balance between these two notions has to be found.

When the iteration number n_i is a multiple of the iteration period i_p , the population dispersion is evaluated based on a statistical metric called the coefficient of variation (CV), also known as the Relative Standard Deviation (RSD) [87]. The coefficient of variation takes into account the fitness values $Fit(\pi)$ of each schedule π belonging to the populations P_0 and P_1 . It is defined as in Eq. 5.14.

$$CV = \frac{\sigma}{\mu} \times 100\% \quad (5.14)$$

$$\text{with } \mu = \frac{1}{2N_{pop}} \sum_{\pi \in P_0 \cup P_1} \frac{1}{Fit(\pi)} \quad (5.15)$$

$$\text{and } \sigma = \sqrt{\frac{1}{2N_{pop}} \sum_{\pi \in P_0 \cup P_1} \frac{1}{(Fit(\pi) - \mu)^2}} \quad (5.16)$$

According to the value of CV, different decisions are made either to enhance the diversification or to intensify the research in some regions of the search space. Three different cases can occur:

1. $CV \leq \varepsilon_{min}$ where ε_{min} is a fixed minimum dispersion threshold
2. $CV \geq \varepsilon_{max}$ where ε_{max} is a fixed maximum dispersion threshold
3. $\varepsilon_{min} < CV < \varepsilon_{max}$

In case 1, the individuals among the populations P_0 and P_1 are very similar and describe only limited regions of the whole search space. Due to this similarity between the individuals, there is a risk that the algorithm converges too fast towards a local optimal solution. To reduce the risk of premature convergence, we apply the receptor editing operator, defined by Ladj et al. [87] with some slight modifications to adapt to our problem and our individual shape. A part of the least fitted individuals (a% of the population) among P_0 and P_1 are eliminated and replaced by new random individuals to increase the possibility to explore other new regions of the research space.

In case 2, the individuals cover too many distinct research space regions. To promote the most promising regions, i.e. the regions containing the most fitted individuals, we select a part of the least fitted individuals (a% of the population) and replace them by mutations of the most fitted ones. We call this stage the generation of new good solutions (see Figure 5.8).

In case 3, we consider that the value of CV is good enough to avoid any change in the individuals among the populations P_0 and P_1 .

Once the changes among the total population have been made, we can define the population that we be used for the next iteration of the genetic algorithm.

Replacement stage

The replacement stage is the stage where the population for the next iteration is selected among the total population formed by P_0 and P_1 . In the first place, we select a small proportion of the least fitted individuals (b% of the population). Then, we complete it with the most fitted individuals until the next iteration population contains N_{pop} individuals. Adding some of the least fitted individuals is also a way to avoid a premature convergence of the algorithm.

Stopping criterion

The different stages are applied in an iterative way until we satisfy the stopping criterion. Each iteration of the GA is called a generation. In our case, the stopping criterion is based

on a maximum number of generations. After i_{max} generations or iterations, the genetic algorithm stops and a joint schedule for missions and maintenance operations minimizing the total maintenance costs is obtained.

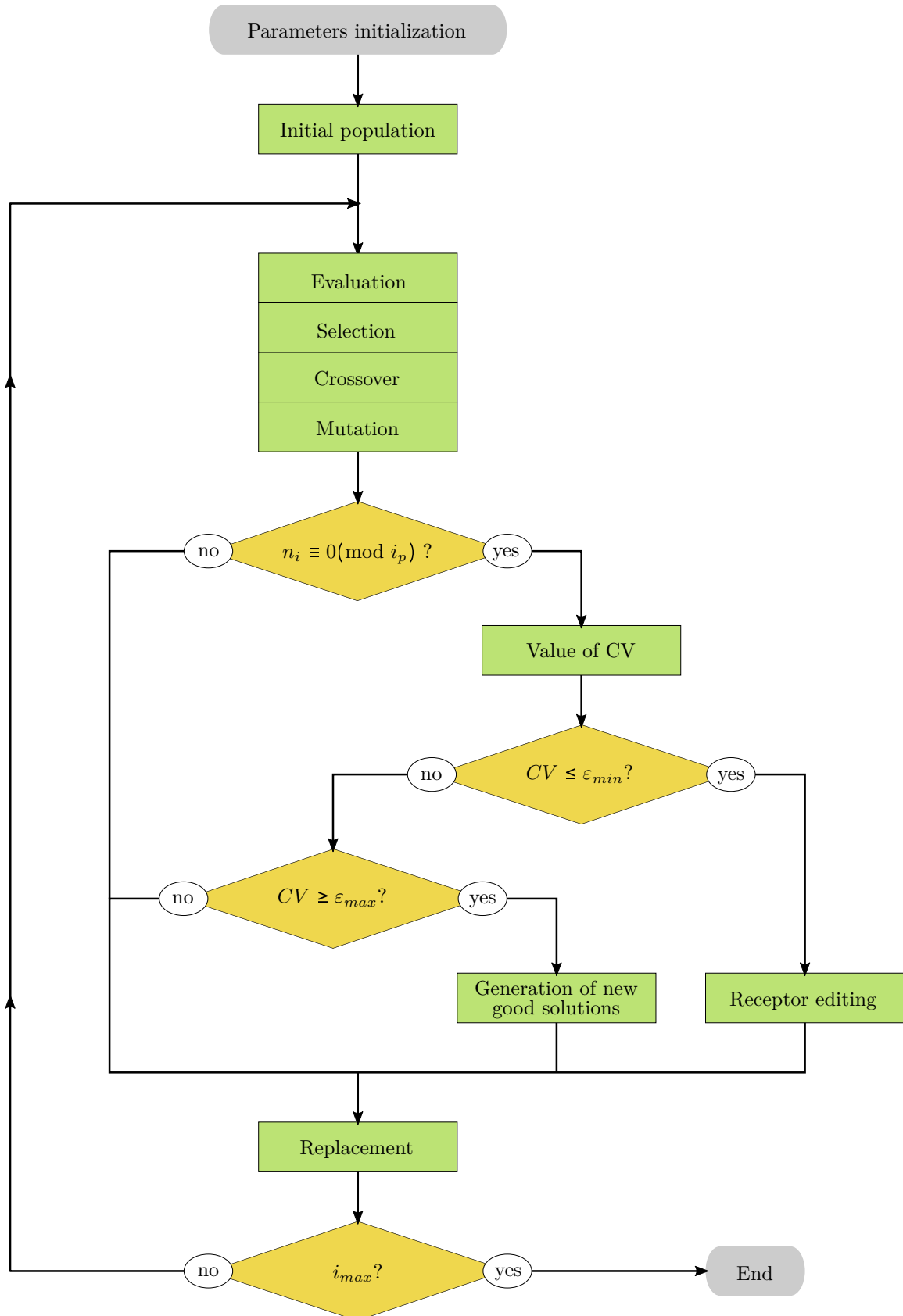


Figure 5.8: Genetic algorithm principle

5.3 Performance of the method

After defining a genetic algorithm to schedule simultaneously the missions and the maintenance operations for a vehicle according to its usage conditions, it is necessary to evaluate the algorithm performances and the effects of different parameters on its behaviour.

These analyses are led through application examples to evaluate the advantages and the limits of the proposed optimization approach and defined genetic algorithm. The performance analysis enables to study the convergence of the genetic algorithm as well as compare the computation time and the final solution with the results obtained thanks to the exact method. A sensitivity study is also led to evaluate the impact of the maximum admissible failure probability \mathbb{P}_{max} that is one of the decision variable of the policy and two parameters:

- the ratio R_c between preventive and corrective maintenance costs respectively denoted C_0 and C_f ,
- the variance of the deterioration on a mission.

5.3.1 Application example

We would like to schedule $n = 6$ missions as well as the necessary maintenance operation for a vehicle.

Let us consider two different datasets A (Table 5.3) and B (Table 5.4). The failure probabilities computed in the tables are obtained by using the failure threshold $L = 100$ and the mission parameters i.e. the duration, α_m and β_m and applying the equation 5.8. Note that the values of the durations, α_m and β_m are chosen for illustrative purposes only. They do not represent values used in real application cases at Volvo.

The dataset A is defined so that the influence of the variances changes for the missions is accentuated while the dataset B enables to better illustrate the sensitivity studies regarding the impact of R_c and the effect of \mathbb{P}_{max} . As the failure probability for each mission in dataset B is higher than in dataset A, it is harder to group the missions for dataset B. The schedule is then more likely to be composed of as many blocks as missions to complete. Increasing the variance of the deterioration induced by the missions adds uncertainty that should lead to increase the number of blocks. But we cannot observe this effect as the schedule has already as many blocks as missions. Dataset A is then better to study the variance variations but dataset B is better to see the evolution of the number of blocks composing the schedule when the constraints defined by R_c and \mathbb{P}_{max} are relaxed. Naturally, the effects of R_c , \mathbb{P}_{max} and the variance variations are intertwined. But these two datasets enable to ease the observations of one another.

Figure 5.9 represents some realizations of the deterioration Gamma processes followed by each mission from dataset A and from the dataset generated based on dataset A but for

which the variance of the deterioration for the missions is multiplied by a factor 5. These realizations are plotted on a longer interval than the mission duration to better illustrate the variance difference.

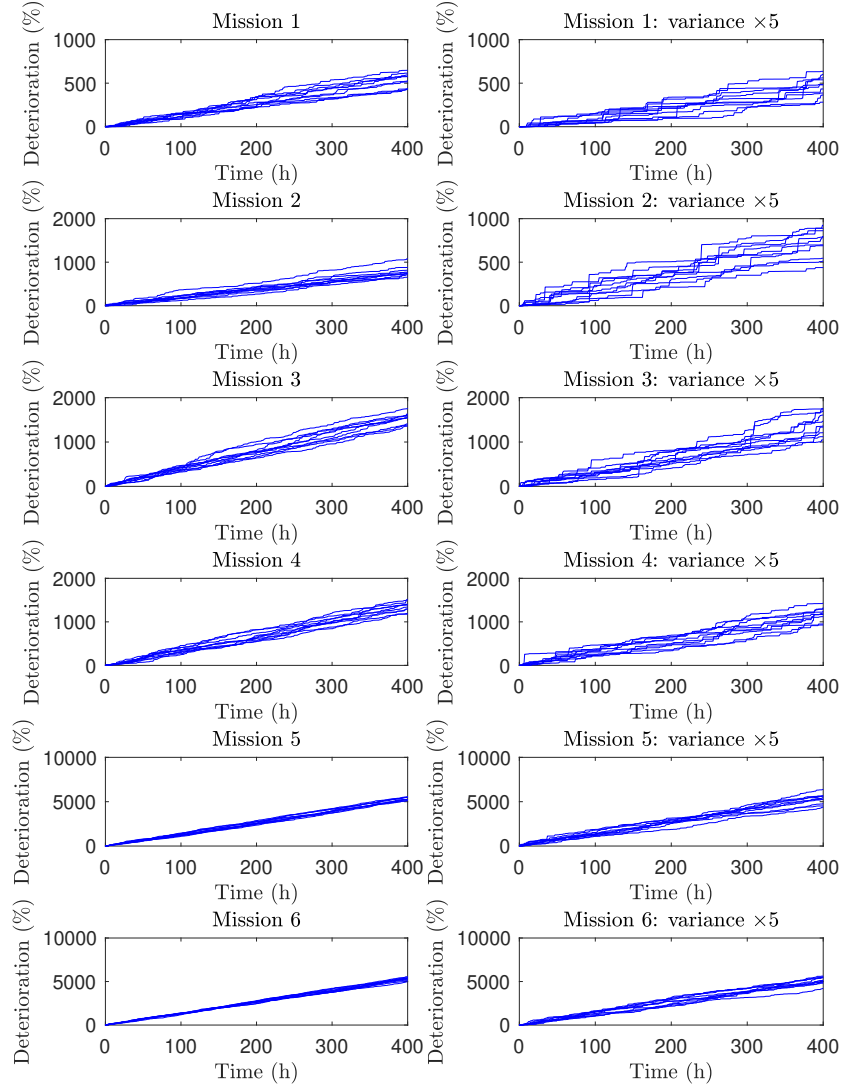


Figure 5.9: Deterioration trajectories for the missions when the variance of the Gamma process has changed

Table 5.5 defines the parameters that have to be initialized for the genetic algorithm as well as the values for the preventive and corrective maintenance costs, respectively denoted C_0 and C_f .

Table 5.3: Dataset A

Missions	Durations(h)	α_m	β_m	Failure probabilities
1	21	0.13	0.1	0.002
2	21	0.18	0.1	0.009
3	8	0.40	0.1	0.004
4	8	0.33	0.1	0.002
5	2	1.33	0.1	0.002
6	3	1.32	0.1	0.01

Table 5.4: Dataset B

Missions	Durations(h)	α_m	β_m	Failure probabilities
1	9	0.85	0.1	0.189
2	4	1.85	0.1	0.163
3	3	1.34	0.1	0.011
4	6	0.93	0.1	0.048
5	6	0.90	0.1	0.041
6	2	3.81	0.1	0.184

Table 5.5: Parameters definition

Parameters	Values	Parameters	Values
C_0	1000	C_f	3000
N_{pop}	30		
P_{cross}	0.7	P_{mut}	0.1
i_p	4	i_{max}	100
a	20%	b	20%
ε_{min}	10	ε_{max}	60

The maximum admissible failure probability \mathbb{P}_{max} for the blocks is fixed at 0.95 for all studies except for the one studying the variance changes impact for which \mathbb{P}_{max} is equal to 0.1. $\mathbb{P}_{max} = 0.1$ means that we have at the maximum a 10% risk to have a failure in a block of missions. As we study a problem with a small number of missions, we choose a high value of \mathbb{P}_{max} . A high value of \mathbb{P}_{max} is chosen to increase the number of possible schedules with the exact method so that we can better compare the performances between the GA and the exact method. For the variance changes study, the chosen value of \mathbb{P}_{max} will be further justified in [Section 5.3.3](#).

5.3.2 Performance analysis

The performance analysis is realized for both datasets and both criteria C_1 and C_2 defined in Section 5.2.2. For each dataset, we apply 1000 times both the genetic algorithm (GA) and the exact method (ExM). For ExM, the only interest of all these simulations is to study the computation time. The simulations are used to compare the obtained results in terms of optimal schedule, maintenance cost criterion value associated to this schedule C_m and computation time T_c . The performance study results are summed up in Table 5.6.

Table 5.6: Performance results

	Dataset A		Dataset B	
Criterion	C_1	C_2	C_1	C_2
C_m	3787.2	3788.3	7898.4	7905
GA: T_c (s)	2.34	2.90	3.02	3.24
EM: T_c (s)	7.79	7.76	3.47	3.48

Dataset A

The optimal joint schedule obtained with the exact method is similar for both decision criteria C_1 and C_2 and is defined in Eq. 5.17. The number of possible joint schedules while respecting $\mathbb{P}_{max} = 0.95$ is equal to 199.

$$\pi_{opt_A}^{1,2} = \{ [1, 3] [2, 5] [4, 6] \} \quad (5.17)$$

The genetic algorithm enables to converge towards the same schedule as the exact method for both criteria for all the 1000 tests. In addition, the computation time gains obtained with the genetic algorithm are quite significant event with such a small study case with only $n = 6$ missions to schedule. For the criterion C_1 and C_2 , we manage to save respectively 70% and 62.6% of the computation time when using the GA (Figure 5.10).

To check the validity of the maintenance cost based criterion, we study the evaluation of the maintenance costs by simulating the schedule obtained with the genetic algorithm to compare it with the optimal value of the criterion. The simulation framework is defined through 5000 scenarios. Each scenario corresponds to a realization of the deterioration process for each set of parameters characterizing a mission. There are then n deterioration trajectories by scenario: one for each mission. The results are presented in Figure 5.11 for the two criteria respectively for dataset A. As the optimal schedule obtained with the genetic algorithm are the same for both criteria, the two graphs are quite similar in shape. The error between the average simulation maintenance costs and the optimal maintenance costs estimated with the criteria C_1 and C_2 are respectively equal to 0.74% and 0.78%.

Dataset B

For dataset B, the optimal joint schedule obtained with the exact method slightly differs

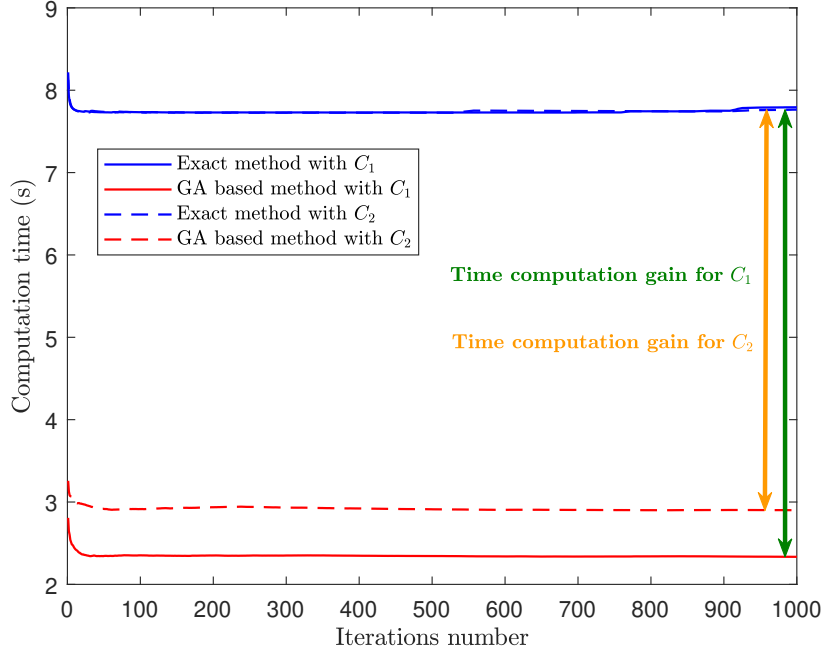


Figure 5.10: Computation time evolution for the dataset A

from one criterion to the other. The obtained schedules for C_1 and C_2 respectively correspond to Eq. 5.18 and Eq. 5.19. In this case, the number of possible schedules respecting the block filling condition $\mathbb{P}_{max} = 0.95$ reaches 80. It explains why the exact method is faster when considering the dataset B (Table 5.6).

$$\pi_{opt_B}^1 = \{ [1][2][4][6][3,5] \} \quad (5.18)$$

$$\pi_{opt_B}^2 = \{ [1][2][3][4][5][6] \} \quad (5.19)$$

The genetic algorithm also converges towards the same solutions for each criterion respectively. However, the computation time gains are lower for this dataset. Indeed, they are respectively of 13.1% for C_1 and of 7% for C_2 (Figure 5.12).

Figure 5.13 shows the convergence of the maintenance costs when simulating the schedule obtained with the genetic algorithm through 5000 different deterioration scenarios. The error between the average simulation maintenance costs and the optimal maintenance costs estimated with the criteria C_1 and C_2 are respectively equal to 0.27% and 0.24%. These errors are smaller than for dataset A insofar as the schedules for each criterion are composed of more blocks. There are then less failure occurrences in each simulation scenario, so less deviation with respect to the optimal maintenance costs estimated with the criteria.

For all the 1000 tests for both datasets and both criteria, the genetic algorithm converges towards the same schedule as the one obtained with the exact method. The only possible

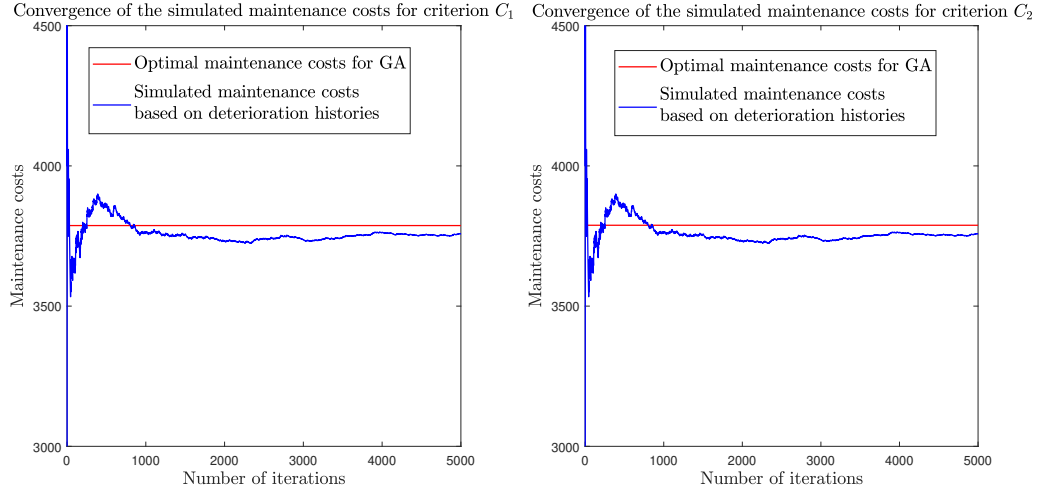


Figure 5.11: Analysis of the maintenance costs convergence towards the optimal value for criteria C_1 and C_2 for dataset A

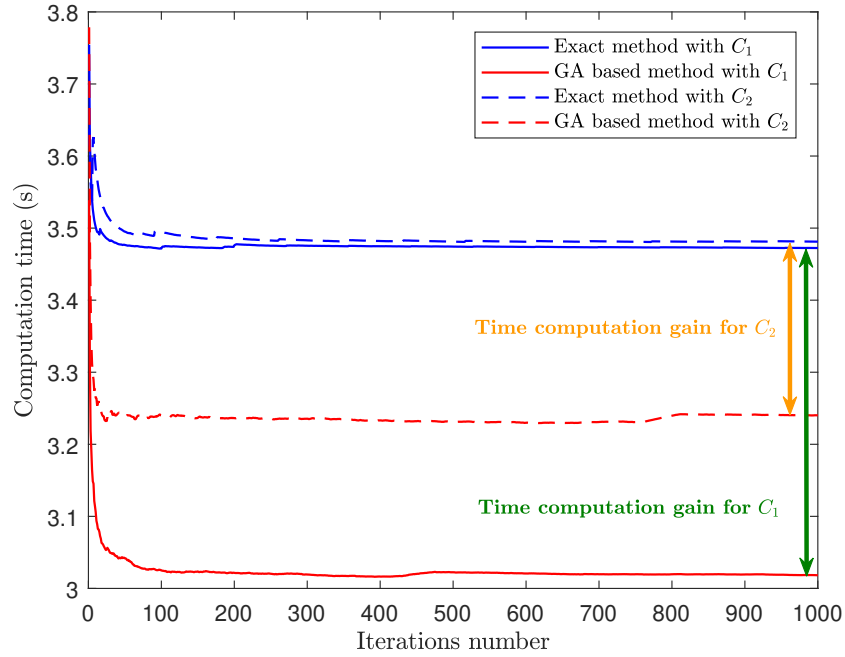


Figure 5.12: Computation time evolution for the dataset B

differences that can occur are permutations either between the blocks or between missions inside the same block. But, as the criteria are evaluated based on the equivalent Gamma process for each block, the mission order does not have any influence on the blocks failure probabilities estimation.

We can notice that the evaluation of the criterion C_2 is slightly higher for both datasets.

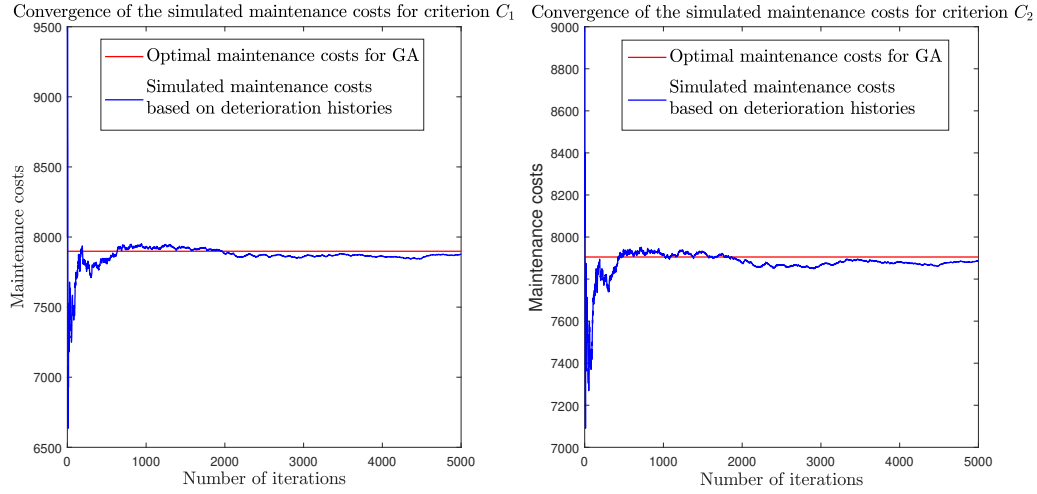


Figure 5.13: Analysis of the maintenance costs convergence towards the optimal value for criteria C_1 and C_2 for dataset B

It is due to the fact that we consider multi failures in the blocks, so it increases the associated corrective maintenance cost. Moreover, the GA computation time for C_2 is higher than the one for C_1 . These increases are about 24% and 7.4% for datasets A and B respectively. Indeed, as C_2 needs to estimate multi failure probabilities, there are more calculations to do when computing the fitness values for the candidate schedules in every iteration.

The obtained results regarding the genetic algorithm performances show the interest of using such a method to find an optimal joint schedule for missions and maintenance operations. Indeed, for small-size problem at least, the genetic algorithm enables to reach the same results as an exact method but also save some computation time. For bigger-size problem, to ensure the algorithm convergence, some adaptation of the GA parameters might be needed. With these promising results, it seems essential to study in details the GA sensitivity regarding some parameters that may influence the obtained schedule definition.

5.3.3 Sensitivity study

In this section, we are going to study the influence of some parameters on the behaviour of the genetic algorithm and the changes that can be aroused on the obtained joint schedule and on the global algorithm performance.

5.3.3.1 Impact of the ratio between the preventive and corrective maintenance costs

The first parameter we are interested in is R_c , the ratio between the preventive and corrective maintenance costs, respectively denoted C_0 and C_f . The value of R_c varies between 0.1 and

1 with a step fixed at 0.1. We keep the value of the preventive maintenance cost C_0 at 1000. The results are described in Table 5.7, Table 5.8, Table 5.9 and Table 5.10 for each dataset and each criterion.

Table 5.7: Effect of R_c when considering the dataset A and the criterion C_1

R_c	C_f	C_1	Final schedule
0.1	10000	5197.3	$\{[4, 3][5, 1][6][2]\}$
0.2	5000	4312.1	$\{[6, 4][3, 1][2, 5]\}$
0.3	3333	3874.7	$\{[5, 2][1, 3][6, 4]\}$
0.4	2500	3656	$\{[1, 3][6, 4][5, 2]\}$
0.5	2000	3524.8	$\{[4, 6][1, 3][2, 5]\}$
0.6	1667	3330.1	$\{[5, 6, 1][2, 3, 4]\}$
0.7	1428.6	3140.1	$\{[4, 3, 2][6, 5, 1]\}$
0.8	1250	2997.5	$\{[2, 4, 3][6, 1, 5]\}$
0.9	1111	2886.7	$\{[2, 3, 4][6, 5, 1]\}$
1	1000	2798	$\{[4, 2, 3][6, 1, 5]\}$

Table 5.8: Effect of R_c when considering the dataset A and the criterion C_2

R_c	C_f	C_2	Final schedule
0.1	10000	5198.1	$\{[4, 3][6][2][1, 5]\}$
0.2	5000	4313.9	$\{[5, 2][6, 4][3, 1]\}$
0.3	3333	3875.9	$\{[5, 2][6, 4][1, 3]\}$
0.4	2500	3656.9	$\{[1, 3][5, 2][6, 4]\}$
0.5	2000	3525.6	$\{[6, 4][3, 1][5, 2]\}$
0.6	1667	3341.4	$\{[1, 5, 6][2, 3, 4]\}$
0.7	1428.6	3149.7	$\{[3, 4, 2][5, 6, 1]\}$
0.8	1250	3006	$\{[2, 3, 4][5, 6, 1]\}$
0.9	1111	2894.2	$\{[6, 5, 1][3, 2, 4]\}$
1	1000	2804.8	$\{[6, 5, 1][2, 3, 4]\}$

Table 5.9: Effect of R_c when considering the dataset B and the criterion C_1

R_c	C_f	C_1	Final schedule
0.1	10000	12350	$\{[1][3][4][6][5][2]\}$
0.2	5000	9175	$\{[3][2][6][4][5][1]\}$
0.3	3333	8116.7	$\{[5][3][1][2][6][4]\}$
0.4	2500	7415.4	$\{[5, 3][4][6][2][1]\}$
0.5	2000	6901.5	$\{[1][6][2][5, 4, 3]\}$
0.6	1667	6345.4	$\{[1, 6][2][4, 3, 5]\}$
0.7	1428.6	5867.4	$\{[4, 5, 3][6, 1][2]\}$
0.8	1250	5509	$\{[5, 3, 4][2][6, 1]\}$
0.9	1111	5230.2	$\{[6, 1][2][5, 3, 4]\}$
1	1000	5007.2	$\{[5, 3, 4][2][1, 6]\}$

Table 5.10: Effect of R_c when considering the dataset B and the criterion C_2

R_c	C_f	C_2	Final schedule
0.1	10000	12350	$\{[4][3][6][1][5][2]\}$
0.2	5000	9175	$\{[6][2][5][4][3][1]\}$
0.3	3333	8116.7	$\{[3][5][1][2][4][6]\}$
0.4	2500	7422.9	$\{[4][5, 3][6][1][2]\}$
0.5	2000	6938.3	$\{[4][2][6][1][5, 3]\}$
0.6	1667	6592.3	$\{[6][3, 4, 5][1][2]\}$
0.7	1428.6	6187.6	$\{[2][3, 4, 5][6, 1]\}$
0.8	1250	5789.2	$\{[2][3, 4, 5][1, 6]\}$
0.9	1111	5479.3	$\{[1, 6][2][4, 3, 5]\}$
1	1000	5231.3	$\{[4, 5, 3][1, 6][2]\}$

For both datasets and both criteria, the number of blocks composing the schedule decreases when R_c increases. Indeed, when R_c increases, the value of the cost for a corrective maintenance C_f is getting closer to the value of a preventive maintenance C_0 . It then becomes less profitable to have a lot of blocks in the schedule.

When the value of R_c is close to 0.1, it means that a corrective maintenance operation is ten times more expensive than a preventive maintenance. Grouping missions increases the failure risk in each block and leads to a maintenance cost inflation. As we try at best to avoid potential failures and to minimize the maintenance cost, it is more relevant to have a final schedule composed of many blocks with few missions inside them.

On the contrary, when the value of R_c is close to 1, the cost for a corrective and preventive maintenance is the same. Grouping missions in a small number of blocks is then less expensive.

Note that the reduction of the number of blocks can be slightly faster for the criterion C_1 (Table 5.9 and Table 5.10 for dataset B). When R_c is equal to 0.5, the optimal schedule for C_1 is made up with 4 blocks while the one for C_2 has 5 blocks.

5.3.3.2 Effect of the variation of the block filling condition

This part study the influence of the block filling condition defined by the maximum admissible failure probability \mathbb{P}_{max} for each block of missions. This probability aims at guiding the genetic algorithm so that it converges faster towards the optimal solution. The smaller the value of this probability is, the stricter the constraint on the way to fill the blocks is. It actually is a leverage to adjust the ratio between the number of preventive and corrective maintenance operations.

For both datasets and both criteria, we study the evolution of the schedule composition and its associated value for the criterion when \mathbb{P}_{max} varies. For dataset A, \mathbb{P}_{max} varies between 0.05 and 1 with a step fixed at 0.05 while for dataset B, the variations are between 0.2 and 1 with the same step. The results are given in Table 5.11 to Table 5.14.

Table 5.11: Effect of \mathbb{P}_{max} when considering the dataset A and the criterion C_1

\mathbb{P}_{max}	C_1	Final schedule
0.05	5185.3	$\{[3][1][6][5,4][2]\}$
0.1	4359.2	$\{[4,3][2][5,1][6]\}$
$\llbracket 0.15; 0.95 \rrbracket$	3787.2	$\{[6,4][2,5][3,1]\}$
1	3787.2	$\{[5,2][6,4][3,1]\}$

Table 5.12: Effect of \mathbb{P}_{max} when considering the dataset A and the criterion C_2

\mathbb{P}_{max}	C_2	Final schedule
0.05	5185.4	$\{[1][6][2][5,4][3]\}$
0.1	4359.4	$\{[6][4,3][1,5][2]\}$
$\llbracket 0.15; 0.95 \rrbracket$	3788.3	$\{[5,2][6,4][1,3]\}$
1	3788.3	$\{[6,4][1,3][2,5]\}$

Table 5.13: Effect of \mathbb{P}_{max} when considering the dataset B and the criterion C_1

\mathbb{P}_{max}	C_1	Final schedule
$\llbracket 0.2; 0.35 \rrbracket$	7905	$\{[5][6][2][3][1][4]\}$
$\llbracket 0.4; 0.95 \rrbracket$	7898.4	$\{[1][2][3,5][4][6][2]\}$
1	4000	$\{[5,6,4,2,1,3]\}$

Table 5.14: Effect of \mathbb{P}_{max} when considering the dataset B and the criterion C_2

\mathbb{P}_{max}	C_2	Final schedule
0.2	7905	$\{[3][4][1][6][5][2]\}$
$\llbracket 0.25; 0.95 \rrbracket$	7905	$\{[6][2][3][1][5][4]\}$
1	7905	$\{[3][1][2][5][4][6]\}$

The global observation we can make is that when the value of \mathbb{P}_{max} increases, the number of blocks composing the schedule tends to decrease. For dataset B, when considering C_1 and \mathbb{P}_{max} equal to 1, the optimal schedule groups all the missions in the same block. It comes from the fact that, as only one failure by block is considered, the failure probability is equal to 1 and cannot overstep this value. In addition, there is no other schedule giving such a low

value for C_1 . However, when \mathbb{P}_{max} is equal to 1, the final schedule is not always composed of a unique block. For dataset A, the final schedule is composed of three blocks when considering the criterion C_1 .

For dataset A, with the same conditions, the value of C_1 for a schedule composed of one block is higher than the one for the obtained schedule. It is harder to reduce the number of blocks with C_2 because multi failure probabilities are considered (Table 5.14). Considering the possibility to have more than one failure in each block is then like adding a constraint to solve the joint scheduling problem. In the case where there are high probabilities to have multi failures in the blocks it is more relevant to use C_2 or we can use C_1 but with a smaller value of \mathbb{P}_{max} , as applying the genetic algorithm with C_2 as decision criterion takes more computation time.

5.3.3.3 Influence of the missions variance changes

We are interested in studying the influence of the changes in the variance of the deterioration for the missions to evaluate the algorithm behaviour. The initial dataset considered for this study is the dataset A. Based on this missions dataset, three other datasets are generated.

Table 5.15: Datasets when increasing the variance by 2 or 5 and when decreasing the variance by 2

Missions	Variance x2			Variance x5			Variance /2		
	α_m	β_m	\mathbb{P}	α_m	β_m	\mathbb{P}	α_m	β_m	\mathbb{P}
1	0.07	0.05	0.02	0.03	0.02	0.05	0.27	0.2	4.2×10^{-5}
2	0.09	0.05	0.04	0.04	0.02	0.09	0.37	0.2	5.8×10^{-4}
3	0.20	0.05	0.02	0.08	0.02	0.07	0.79	0.2	1.1×10^{-4}
4	0.16	0.05	0.01	0.07	0.02	0.05	0.66	0.2	2.5×10^{-5}
5	0.66	0.05	0.01	0.27	0.02	0.05	2.65	0.2	2.7×10^{-5}
6	0.66	0.05	0.04	0.26	0.02	0.09	2.64	0.2	7.2×10^{-4}

The durations of the missions are the same but the deterioration processes parameters are modified so that the expected value for the law followed by the deterioration increments of each mission remains the same from one dataset to another. For each mission, the variance of the deterioration increment is either increased, i.e. multiplied by 2 or 5, or decreased, i.e. divided by 2, with respect to the initial variance value obtained with the deterioration parameters from the dataset A. Table 5.15 shows the different parameters values for the new datasets with variance changes. \mathbb{P} denotes the mission failure probability. The study is led for two values of \mathbb{P}_{max} : 0.95 and 0.1.

Study when $\mathbb{P}_{max} = 0.95$

With both criteria, the optimal schedule for dataset A is $\pi_{opt_A}^{1,2} = \{[1, 3][2, 5][6, 4]\}$. When increasing the variance by a factor 2, 5 or when decreasing the variance by a factor 2, the schedules obtained for the both criteria are respectively given by Eq. 5.21, 5.22 and 5.20.

$$\pi_{opt_{j2}}^1 = \{ [3, 1] [2, 5] [4, 6] \} \quad \pi_{opt_{j2}}^2 = \{ [3, 1] [2, 5] [4, 6] \} \quad (5.20)$$

$$\pi_{opt_{x2}}^1 = \{ [3, 5, 1, 4] [6, 2] \} \quad \pi_{opt_{x2}}^2 = \{ [3, 1] [6, 4] [5, 2] \} \quad (5.21)$$

$$\pi_{opt_{x5}}^1 = \{ [6, 3, 1, 2, 4, 5] \} \quad \pi_{opt_{x5}}^2 = \{ [5, 1, 4] [2, 6, 3] \} \quad (5.22)$$

It seems that when increasing the variance values, the number of blocks composing the schedule decreases. It can be surprising because when the variance increases, the uncertainty regarding the failure occurrence increases. It should then lead to increase the number of blocks when increasing the variance.

To better understand this phenomenon, let us study more in details what happens for the criterion C_1 when the variance increases. Table 5.16 presents the obtained schedules with the failure probabilities for the blocks as well as the criterion values.

Table 5.16: Schedules obtained with the criteria C_1 and C_2

Dataset	Final schedule	Blocks failure probabilities	C_1
A	$\{ [1, 3] [2, 5] [6, 4] \}$	$\{ [0.0657] [0.0958] [0.1010] \}$	3493.8
Variance x2	$\{ [3, 5, 1, 4] [6, 2] \}$	$\{ [0.5523] [0.2505] \}$	4408.4
Variance x5	$\{ [6, 3, 1, 2, 4, 5] \}$	$\{ [0.8313] \}$	3493.8

To analyse the obtained results, we consider that all the missions are put in a single block, like the obtained schedule when the variance values are increased by a factor 5. The objective is to plot the distribution of the failure time. We propose a simulation test that generates $N = 10000$ deterioration trajectories based on the equivalent Gamma process parameters for the block composed of the $n = 6$ missions. These trajectories are generated until $D_t = 63$, the total duration of the block. For each trajectory, a value is gathered at every time step $D_t/1000$ and the moment when the threshold $L = 100$ is reached is also retrieved. Thanks to these moments, histograms of the failure distribution and of the probability density can be plotted and compared with the estimations of the failure distribution and the probability density of the failure time based on the equivalent Gamma process. Figures 5.14 to 5.16 represent the comparison between the histograms and the failure distribution and probability density function.

For the dataset A (Figure 5.14) and the dataset with variance x2 (Figure 5.15), the failure probabilities for a unique block filled with all the missions are respectively equal to 0.993 and 0.954. As it is greater than $\mathbb{P}_{max} = 0.95$, such a schedule is not admissible with the missions parameters. However, for the dataset with variance x5, the failure probability for a unique block is equal to 0.831. This value is smaller than \mathbb{P}_{max} . It explains why this schedule is admissible. Figure 5.16 shows that the tail of the probability density function is spread to the right side of the graph when the variance of the failure law increases. It means that the probability to have a schedule with a unique block without having a failure has become different from 0. As this probability is not equal to 0 and is smaller than the maximum

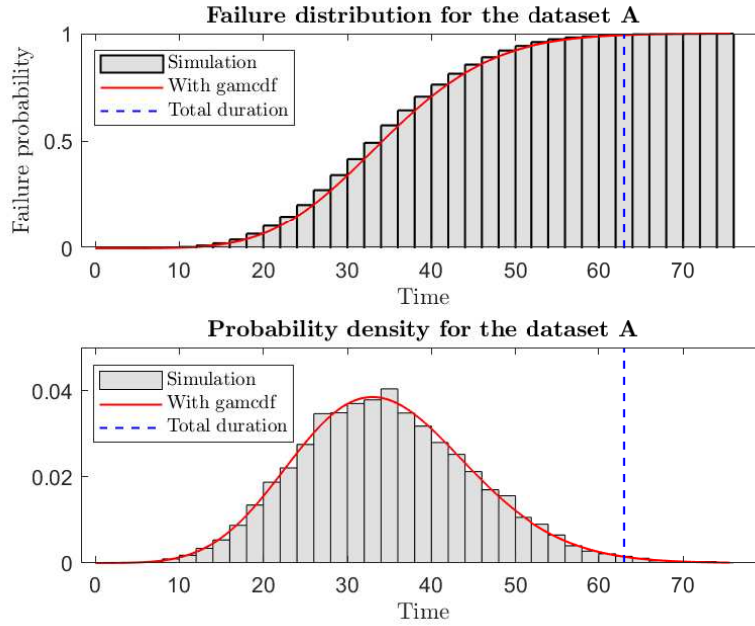


Figure 5.14: Failure distribution and probability density compared with the results from the simulation for dataset A

admissible failure probability, the algorithm takes it into account but does not consider the fact that there may be also more precocious failures.

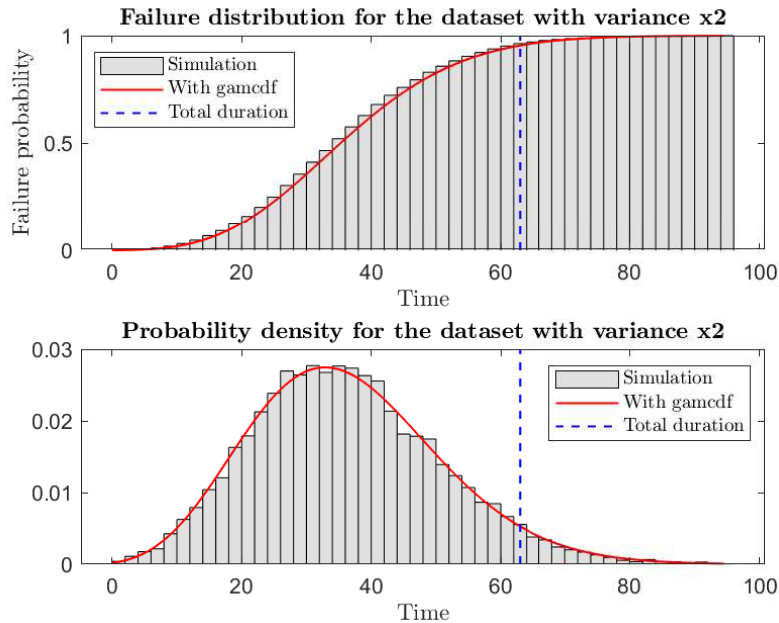


Figure 5.15: Failure distribution and probability density compared with the results from the simulation for the dataset with variance x2

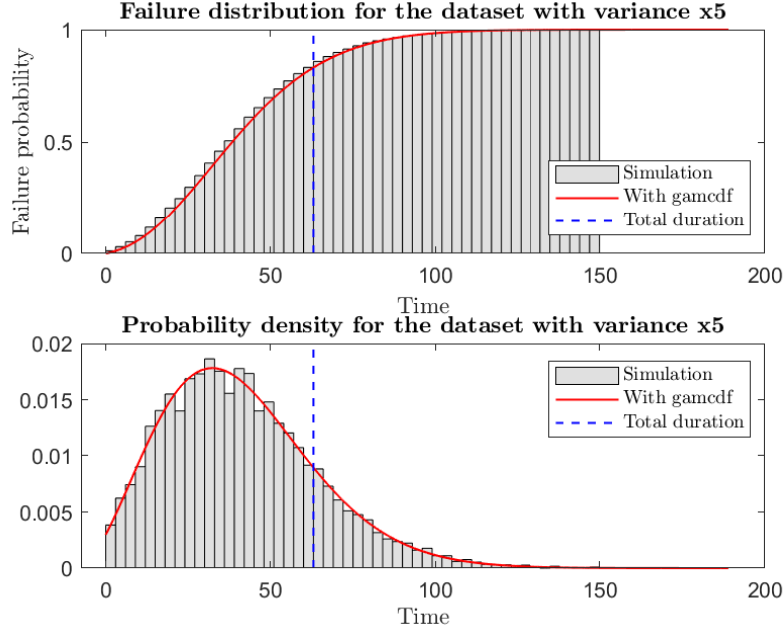


Figure 5.16: Failure distribution and probability density compared with the results from the simulation for the dataset with variance x5

The same phenomenon explains why the optimal schedule $\pi_{opt_{x2}}^1$ for the criterion C_1 is composed of two blocks, one of which is filled with 4 missions. For the criterion C_2 , the effect of the probability density tail still occurs but the reduction of the blocks number composing the schedule is slower. It is due to the fact that we consider the multi failure probabilities for each block. It then affects more the decision criterion than when we consider only the possibility of one failure occurrence by block.

To be sure to only observe the impact of the variance changes, we choose to set \mathbb{P}_{max} at 0.1.

Study with $\mathbb{P}_{max} = 0.1$

The final schedules and their associated values for the criteria C_1 and C_2 are presented in Table 5.17. Increasing the values of the variances increases the number of blocks composing the optimal joint schedule for both criteria. When increasing the variances by a factor 2 or 5, we obtain schedules with respectively 5 and 6 blocks, while for dataset A, we have a schedule composed of 4 blocks. On the contrary, when the variances are reduced by a factor 2, the number of blocks is reduced. The final schedule is composed of 3 blocks. If the variances continue to decrease, the optimal schedule remains $\pi_{opt_{j2}}^1$ for C_1 and $\pi_{opt_{j2}}^2$ for C_2 because reducing more the number of blocks leads to failure probabilities exceeding \mathbb{P}_{max} .

This observation corresponds to what was expected. Indeed, increasing the variance also increases the probability to have one failure for each mission. It then becomes harder to group the missions into blocks without exceeding the maximum admissible failure probability \mathbb{P}_{max} .

The results are explained by the fact that the failure uncertainty increases along with the variance.

Table 5.17: Schedules obtained with the criteria C_1 and C_2

	Criterion C_1		Criterion C_2	
Dataset	Final schedule	C_1	Final schedule	C_2
Variance /2	$\pi_{opt_{j2}}^1 = \{[1, 3][6, 4][5, 2]\}$	3309.4	$\pi_{opt_{j2}}^2 = \{[5, 2][6, 4][3, 1]\}$	3309.4
A	$\pi_{opt_A}^1 = \{[6][5, 1][2][3, 4]\}$	4359.2	$\pi_{opt_A}^2 = \{[6][2][1, 5][3, 4]\}$	4359.4
Variance x2	$\pi_{opt_{x2}}^1 = \{[5, 4][6][1][2][3]\}$	5601.4	$\pi_{opt_{x2}}^2 = \{[4, 5][6][1][2][3]\}$	5606.1
Variance x5	$\pi_{opt_{x5}}^1 = \{[3][6][2][1][4][5]\}$	7199.6	$\pi_{opt_{x5}}^2 = \{[5][6][3][1][2][4]\}$	7199.6

5.3.4 Limitations of the genetic algorithm

The application example is defined for a set of $n = 6$ missions. The first question we can ask is: what will happen for larger size problems ? Indeed, the higher the number of missions to schedule is, the harder it becomes to compare the genetic algorithm results with the exact method ones owing to the exact method computation time and the high risk to reach combinatorial explosion. The number of feasible schedules for the exact method becomes too numerous. For instance, when considering $n = 9$ missions to schedule with $\mathbb{P}_{max} = 0.95$, the exact method needs more than 11 days to reach the optimal schedule. The computation time for the genetic algorithm is much smaller, respectively equal to 2.63s and 3.77s for the criteria C_1 and C_2 . Note that when using C_1 , there are 16% of the 1000 GA running tests that do not reach the optimal schedule and the criterion deviation represents 1.45% of the criterion value for the optimal schedule. Considering several potential failure occurrences by block gives a better maintenance cost estimation and a more coherent surface for the cost criterion from one iteration to another. The convergence is then improved insofar as when using the criterion C_2 , all the obtained schedules for the 1000 tests correspond to the optimal one.

Tuning the different parameters of the genetic algorithm, such as the population size N_{pop} and the maximum iteration number i_{max} , is a way to improve the genetic algorithm convergence. But there is no guarantee that it will work. In addition, if we increase the values of N_{pop} and i_{max} , it will be at the expense of the computation time.

5.4 Conclusion

In this chapter, the static joint scheduling problem has been presented and two methods have been proposed to solve it. The first one is an exact method that test all the feasible schedules respecting the constraints on the block filling. This method is efficient for small size problems

but once the number of missions to schedule increases drastically, the computation time is far too long to reach an optimal schedule. A second method based on a genetic algorithm has then been introduced. Its objective is to reach a schedule close enough to the optimal one in a satisfying computation time. Two maintenance cost based criteria have been defined to optimize the schedule for missions and maintenance operations for a single vehicle.

We present a study based on a small size problem, but even on larger size problems, like 20 missions to schedule or more, the genetic algorithm is still an efficient method to solve the static joint scheduling problem. Naturally, the algorithm will need a longer computation time to converge towards a satisfactory schedule. The computation time is not a constraint in this case because, as we study the static problem, this schedule is never updated. The only changes necessary to study a larger size problem are about the genetic algorithm parametrization. The parameters values need to be optimized to avoid an early convergence towards a local optimum and the stopping criterion also has to be updated to converge in a reasonable time towards a good solution.

Thanks to the performance analysis and the sensitivity study, we see the potential offered by an optimization method based on a genetic algorithm. It gives good first results but also offer improvement perspectives.

Indeed, the static joint scheduling problem is a simplification of the dynamic scheduling problem. Dynamic information such as deterioration measurements, failure occurrences could be integrated to update the joint schedule and reduce the maintenance costs. New missions could also be added during the schedule completion.

In addition, we consider only the maintenance costs as a decision criterion but all the costs associated with the deliveries also have an impact on the way to define the optimal joint schedule. In fact, deliveries are often defined by a deadline that cannot be overstepped. Otherwise, penalty costs are imputed.

In the following chapter, all these different comments are considered to help solving a more complex optimization problem: the dynamic joint scheduling problem for missions and maintenance operations for a single vehicle.

Joint scheduling of missions and maintenance for a vehicle: the dynamic case

Contents

6.1 Problem statement	132
6.1.1 Hypotheses and constraints	132
6.1.2 Objective	133
6.2 Resolution approach description	135
6.2.1 Vehicle deterioration model	136
6.2.2 Decision criteria	136
6.2.2.1 Global maintenance cost estimation	136
6.2.2.2 Production gain estimation	137
6.2.3 Dynamic scheduling algorithm	139
6.2.3.1 Dynamic scheduling principle	139
6.2.3.2 Changes in the genetic algorithm functions	143
6.3 Evaluation of the dynamic scheduling method	144
6.3.1 Rescheduling opportunities: failure occurrences and deterioration information	144
6.3.1.1 Chosen decision criterion for the algorithms	145
6.3.1.2 Cost convergence analysis	145
6.3.1.3 Effect of the rescheduling condition ΔC_{min}	148
6.3.2 Rescheduling opportunities: failure occurrences, deterioration information and new available missions	150
6.3.2.1 Chosen decision criterion for the algorithms	152
6.3.2.2 Cost convergence analysis	152
6.3.2.3 Effect of the rescheduling limit condition ΔC_{min}	156
6.3.3 Rescheduling analysis for different criteria related to the operating incomes estimation	158
6.3.4 Limits of the dynamic scheduling algorithm	160
6.4 Conclusion	161

The previous chapter develops the methodology to define a joint schedule for both missions and maintenance operations for a single vehicle deteriorating over time. This first step enables to start digging into the scheduling problem by reducing the problem complexity to only consider a static study case. Indeed, the schedule is defined based on the assumption that all the missions are known and that no information enables to update this schedule no matter what occurs during its completion. This approach, even if it is a necessary first step, is of course too limited and needs to be extended to the dynamic case.

This chapter proposes a dynamic methodology to jointly schedule missions and maintenance operations by considering different events that can occur during the vehicle lifetime as opportunities to update the initial schedule and fit to the possible usage evolutions and the new operational constraints. There are three different types of information that are taken into account: failure occurrences, deterioration measurements and new missions.

This chapter begins with a definition of the dynamic scheduling problem for a single vehicle and the hypotheses made for this study case. The resolution approach is presented and then evaluated for different application examples to illustrate the value brought by a dynamic rescheduling approach rather than a static one. The first example deals with dynamic rescheduling when considering only failure occurrences and the available deterioration measurements [142] while the second one also adds the possibility to reschedule when new missions are added to the mission list [141].

6.1 Problem statement

This section depicts the dynamic joint scheduling problem for a single vehicle. The huge difference with respect to the static scheduling problem is described to highlight the interest of using the real time information to update the schedule.

6.1.1 Hypotheses and constraints

The hypotheses defined for the dynamic joint scheduling problem are not very different from the ones considered by the static joint scheduling problem. Indeed, we consider the vehicle as having a global health state indicator that enables to follow its deterioration evolution over time, according to the operating conditions and usage changes due to the different missions severity levels. The deterioration phenomenon is then still modelled using a Gamma process.

- **New missions:**

This vehicle has a set of missions to complete and this set can potentially evolve over time with the arrival of new missions to complete. Some features have been added to

characterize the missions. They are still defined by a duration and a usage severity to represent their influence on the deterioration process. We also add some new features more related to the gain that the mission generates for the customers. These features are represented by three parameters:

- $g_m(i)$: the raw gain earned by each mission i ,
- $c_{ud}(i)$: the unitary delay cost that corresponds to the cost charged for each delay unit,
- $d_{s_{max}}(i)$: the starting deadline i.e. the maximum admissible date to start the mission. This feature is very important as it is the one that can force a schedule update to introduce, for instance, a high priority mission.

When a mission has started, it is not possible to stop it to update the schedule if new missions are available. We need to wait for the on-going mission end to integrate the new missions in the schedule.

- **Failure occurrence:**

If a failure occurs, an update is immediately launched because the schedule is not adapted to the vehicle usage anymore.

- **Deterioration information:**

When deterioration information is available, a rescheduling is not mandatory. It only depends on the impact of such information on the current schedule. We assume in the application examples that deterioration information is available at the end of some missions.

- **Maintenance decision:**

The maintenance policy is the same as for the static scheduling problem. When a preventive maintenance operation is scheduled, the cost C_0 is charged while, if a failure occurs during a mission completion, the cost C_f is charged. After every maintenance operation, corrective or preventive, the deterioration state of the vehicle goes back to 0. Preventive maintenance operations are assumed to be less time consuming than corrective maintenance operations insofar as they are scheduled long before their occurrences. Their duration will be considered to schedule at best the preventive maintenance time slots.

6.1.2 Objective

Our objective is to build an adaptive joint schedule for missions and maintenance operations using the information offered by the missions features and the vehicle deterioration model as well as real time information to optimize the predictive maintenance strategy. We then would like to develop a strategy that uses the real time information and the different events to sequentially update the schedule. This is what we call a predictive-reactive strategy [170].

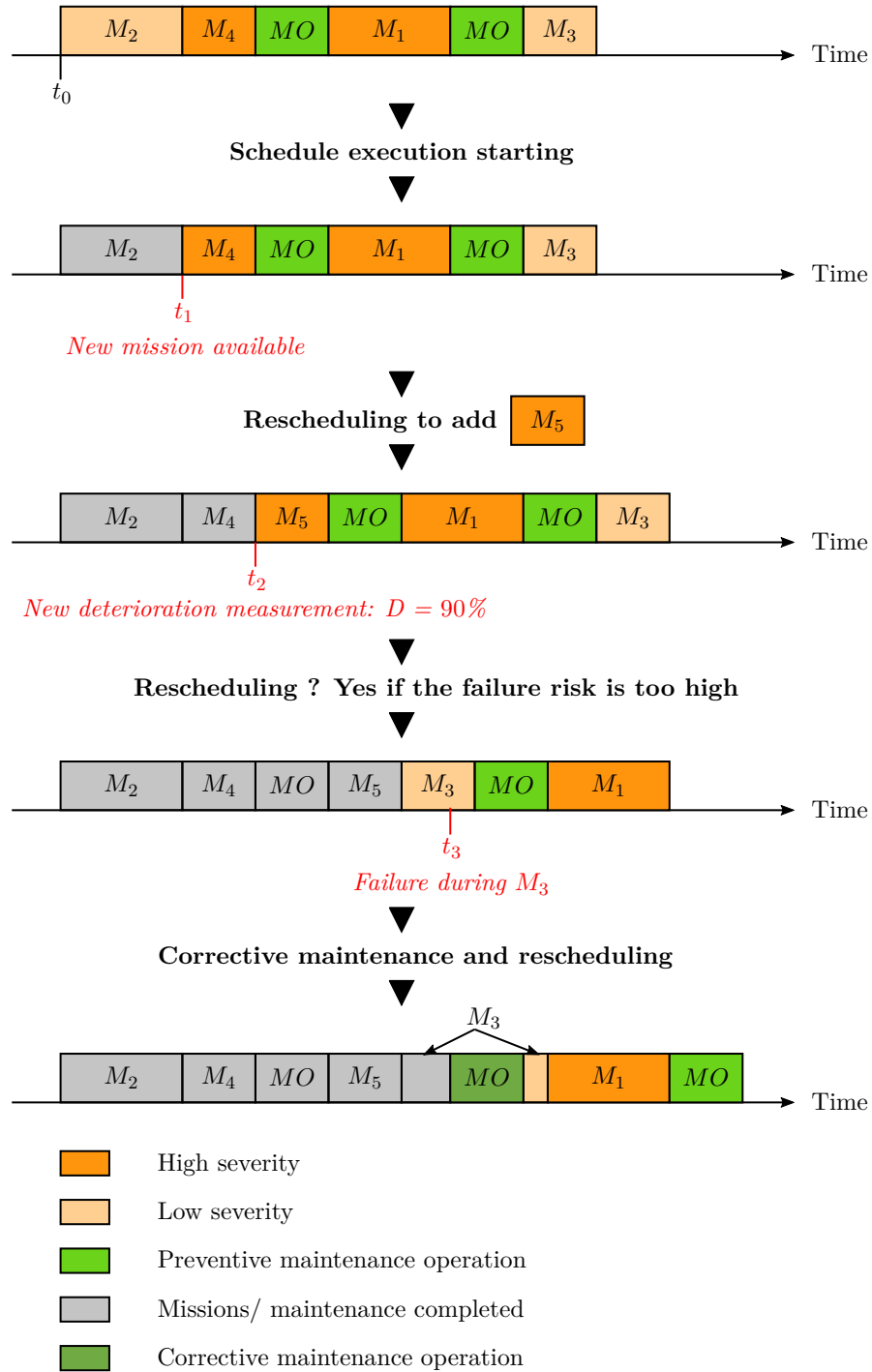


Figure 6.1: Framework to schedule and reschedule the missions according to the occurring events: new mission M_5 , deterioration measure, failure

The main challenges lie in the way to reorganize the missions when new ones have to be

added to the schedule and to evaluate if an event or a piece of information is significant enough to initiate a rescheduling. Indeed, rescheduling means taking the risk of inducing additional logistic costs to re-organize the order of the following missions as well as the maintenance operations time slots.

Figure 6.1 describes the principle applied to schedule the missions and update the initial schedule according to the different events that can occur during the schedule completion. At the beginning, an initial schedule is defined to complete the missions M_1 , M_2 , M_3 and M_4 as well as preventive maintenance operations MO at the right time to fit to the vehicle usage. If there is a new mission to add in the missions batch, like M_5 at t_1 , a rescheduling is applied to update the initial schedule. The mission M_5 will be performed just after M_4 . When deterioration information is available at t_2 , the schedule is not automatically updated. It depends on the vehicle health state and the impact of this measurement on the current schedule. In the presented case, going on with the current schedule means taking a too high risk of failure. It is then relevant to update the schedule. The remaining missions to complete and the maintenance operations are also reordered. Finally, after a failure occurrence during M_3 , a corrective maintenance operation is carried out to repair the vehicle and a rescheduling is applied knowing that the first mission to schedule among the remaining ones is the remaining part of M_3 . The preventive operation that should have been completed after M_3 is then postponed after M_1 .

The schedule optimization can either be based on the maintenance costs or can also consider the gains earned when the different missions are completed. It means that the potential delivery delay has to be considered for each mission to avoid at best delay penalty costs because the hauler commits himself to ensure his customers' deliveries at the right time to be fully paid.

6.2 Resolution approach description

This section develops the methodology applied to schedule and reschedule the missions and the maintenance operations according to the occurring events during the vehicle lifetime. Three subsections enable to describe the process. The first one offers a brief reminder of the vehicle deterioration model. This part is quite similar to the one presented to solve the static joint scheduling (see Section 5.2.1). The second part develops the decision criteria that can be considered to optimize the joint schedule. According to the one we chose, the decision-making process will not lead to the same decisions. The last part describes the dynamic rescheduling method to optimize the initial schedule while taking into account all the available real time information regarding the vehicle state and the operating requirements.

6.2.1 Vehicle deterioration model

We still consider the vehicle has a single-component system whose deterioration is a stochastic process. Based on this assumption, a continuous deterioration process is designed. This process is assumed to be a Gamma process with varying parameters insofar as the vehicle evolves in changing operating conditions due to the different missions it has to complete.

A schedule is composed of blocks of missions separated by preventive maintenance operations. The deterioration model enables to make decisions on how to define these blocks to avoid failures during missions by scheduling the maintenance operations at the right time.

Based on this model, we can estimate the probability to have a failure in the blocks, i.e. the probability to exceed the failure threshold L . Moreover, as monitoring information on the vehicle health state is available during the vehicle lifetime, it is possible to estimate the remaining useful lifetime of the vehicle. We assume that monitoring information can only be available at the end of a mission, but not at the end of every mission. These estimations will help to choose when to schedule maintenance operations, when to deploy the vehicle on missions and how to update the current schedule to consider the deterioration measurements.

The maintenance model, a deterioration-threshold failure model, enables to know how to build the schedule. However, an optimization criterion is necessary to make the best decisions to schedule maintenance operations and missions.

6.2.2 Decision criteria

When scheduling both missions and maintenance operations, we are interested in two different costs: the production gain G generated by the missions achievement and the global maintenance cost C_m associated with preventive and corrective maintenance operations carried out on the vehicle during its lifetime.

6.2.2.1 Global maintenance cost estimation

As described in [Section 5.2.2](#), the global maintenance cost C_m is composed of two parts. The first one corresponds to the maintenance costs associated with preventive maintenance operations competed at the end of every block of missions. The second one is the maintenance cost associated with failure occurrences.

To estimate the corrective maintenance cost, we can either consider one failure by block of missions or several ones. Considering one failure by block for the schedule π means that we estimate C_m as done with the criterion C_1 (Eq. [5.10](#)). When taking into account the possibility to have more than one failure by block, there are different strategies to estimate the cost associated with potential failure occurrences. We can either choose to define a maximum number of possible occurrences, as we did with the criterion C_2 (Eq. [5.11](#)) by

considering that we could not have more failures in a block than the number of missions composing the block. Otherwise, we can consider as many failure occurrences as long as the probability to have n^{th} failures does not exceed a fixed value. In this case, the maintenance cost based criterion can be defined as in Eq. 6.1 to estimate the global maintenance cost for a schedule π composed of N_b blocks. $N(k)$ is the maximum number of failure occurrences admissible in block k and $\mathbb{P}_f(k, i)$ is the probability to exceed the failure threshold L for the i^{th} time for the block k . We assume that as long as $\mathbb{P}_f(k, i)$ is greater than 1%, $N(k)$ is incremented.

$$C_3(\pi) = \sum_{k=1}^{N_b} \left(C_0 + C_f \sum_{i=1}^{N(k)} \mathbb{P}_f(k, i) \right) \quad (6.1)$$

where C_0 and C_f are respectively the preventive and corrective maintenance costs. C_f is assumed to be greater than C_0 to include the immobilization costs and the cost charged by the breakdown service.

6.2.2.2 Production gain estimation

The production gain is assumed to be composed of two parts: the raw gain generated by each completed mission and the delay cost associated with a delay in the delivery. We use the criterion G_p to estimate the gain generated by a schedule π composed of n missions. It is described in Eq. 6.2.

$$G_p(\pi) = \sum_{i=1}^n (g_m(i) - c_d(i)) \quad (6.2)$$

where $g_m(i)$ and $c_d(i)$ are the raw gain and the delay cost for mission i . The production gain for mission i is then equal to $g_m(i) - c_d(i)$.

To evaluate the cost associated to the potential delay for each mission, some assumptions have been made to simplify the estimations of the missions starting deadlines and avoid spending too much computation time on it. Simplifying these estimations is possible insofar as the log frequency of the monitoring information is high enough to ensure that the estimation error for the starting deadlines does not have a too significant impact on the decision-making process to define the schedule. Indeed, as the updates are frequent, the obtained schedule often evolves over time.

The delay cost estimation for mission i , denoted $c_d(i)$, is based on the unitary delay cost $c_{ud}(i)$ and the estimated delay time $t_d(i)$ for this mission (Eq. 6.3). The delay time is supposed to stand for the time difference between the starting deadline of the mission $d_{s_{max}}(i)$ and the real date $t_s(i)$ when the mission starts during the schedule completion (Eq. 6.4).

$$\forall i, \quad c_d(i) = t_d(i) \times c_{ud}(i) \quad (6.3)$$

$$\forall i, \quad t_d(i) = \max\left\{0; t_s(i) - d_{s_{max}}(i)\right\} \quad (6.4)$$

The real starting date $t_s(i)$ for mission i is the hardest part to estimate. Indeed, it is necessary to know the number of preventive maintenance operations completed before the mission execution and the number of failures that occurred. It enables to estimate the time spent in maintenance before the mission execution. It is also required to know which missions have been completed before to estimate the time spent on missions. With that information, the real starting date can be estimated.

The mission i belongs to the k^{th} block in the schedule. The first stage is to estimate the expected time $\mathbb{E}[t_s(k)]$ when the block k containing the mission i begins (Eq. 6.5). The time spent for preventive maintenance is based on the number of previous performed preventive maintenance operations, equal to $(k - 1)$, and the fact that each one lasts d_p time units. The time spent in missions corresponds to the time necessary to complete all the missions in the $k - 1$ previous blocks and can be computed by considering the associated duration for each mission j , $d(j)$. Finally, to estimate the time spent for corrective maintenance, the average number of failures in the $k - 1$ previous blocks has to be assessed and multiply by the duration d_c of a corrective maintenance operation.

$$\mathbb{E}[t_s(k)] = t_0 + (k - 1)d_p + \sum_{b=1}^{k-1} \left(d_c \mathbb{E}[N_f(b)] + \sum_{j \in B_b} d(j) \right) \quad (6.5)$$

where t_0 and $\mathbb{E}[N_f(b)]$ respectively denote the initial time when the schedule starts and the average number of failures in the block b .

The second stage consists in studying what is happening in the block k containing the mission i . We need to consider the durations of the missions completed before starting the mission i as well as the average number of failures that can occur during this period. It enables to estimate $t(k, i)$ the time spent in the block k before starting the mission i (Eq. 6.6).

$$t(k, i) = d_c \mathbb{E}[N_f(k)] + \sum_{j \in B_k \mid M_j < M_i} d(j) \quad (6.6)$$

The global production gain G can be estimated by joining the two stages. It defines the criterion C_4 for the schedule π composed of n missions to estimate G (Eq. 6.7).

$$C_4(\pi) = \sum_{i=1}^n \left(g_m(i) - c_{ud}(i) \times \max\left\{0; \mathbb{E}[t_s(\{k \mid i \in k\})] + t(\{k \mid i \in k\}, i) - d_{s_{max}}(i)\right\} \right) \quad (6.7)$$

where mission i is in block k .

Considering C_4 enables to have a good estimation of the global production gain but can be quite time consuming to compute. Moreover, if monitoring information is often available and lead to many rescheduling occurrences, the impact of the criterion estimation on the global computation time can impact the reactivity of the dynamic scheduling method.

To avoid such a problem, it is possible to reduce the estimation of the starting date $t_s(i)$ for each mission i to $\mathbb{E}[t_s(k)]$ the average time when the block k containing the mission i begins. It means that we are deliberately introducing an estimation error. However, an hauler does not know all the missions that he will have to complete for more than a couple of months and deterioration information could be retrieved at least every week to update the schedule. This estimation error will then not have a huge impact on the quality of the decision-making process to optimize the schedule. If we even want to reduce the estimation error, we can just remove the estimation of $\mathbb{E}[N_f(k)]$, the average number of failures that can occur during the missions in block k completed before starting mission i .

6.2.3 Dynamic scheduling algorithm

The dynamic scheduling algorithm aims at populating the blocks of missions and schedule the preventive maintenance at the right time to optimize a chosen criterion. In our case, we consider a cost based criterion that can either consider the global maintenance cost or the production gain or a mix of both.

6.2.3.1 Dynamic scheduling principle

The initial schedule (Figure 6.2) is generated using the static scheduling algorithm developed in Chapter 5. The missions originally available are scheduled as well as the preventive maintenance operations to optimize the decision criterion. Dynamic scheduling implies schedule updates when monitoring information has to be considered. The way to take into account that information is described in this section.

During the initial schedule implementation and execution, sequential rescheduling can then occur based on three different types of events: a failure occurrence during a mission, a real-time deterioration measurement is available or new missions are requested and added to the current mission list. The decision-making process to rather update or not the current schedule is run at the end of each mission as long as the number of completed missions does not reach the total number of missions to schedule N_t (Figure 6.2). Of course, N_t is incremented every time a new mission is requested.

The different real-time events trigger possible schedule updates:

- **A failure occurs during a mission:** A corrective maintenance operation is performed to restore the vehicle health state to an as good as new state. It incurs a cost C_f . Corrective maintenance is a major event to consider when scheduling the next missions

and the preventive maintenance time slots as it strongly affects the current vehicle health state. It is then mandatory to initiate a rescheduling on the remaining missions. In addition, the first mission to schedule is the undone part of the unfinished mission during which the failure occurs. An adaptation of the static algorithm described in [chapter 5](#) is then implemented to take into account this constraint and to update the current schedule.

- **A new mission is requested:** This mission is added to the mission list. It has to be scheduled as soon as possible to avoid any possible delay cost due to the starting deadline characterizing the mission. As a mission cannot be stopped once it has started, a new mission can only be considered at the end of the on-going mission. A mandatory rescheduling is then triggered as soon as the current mission is completed to integrate the new mission in the current schedule. This rescheduling affects the remaining missions as well as the already scheduled maintenance time slots.
 - ◆ If a deterioration measurement is available at the end of the completed mission just before the rescheduling order, it is integrated in the rescheduling prerequisite. It will condition the way to fill in the first block composing the new schedule.
 - ◆ If there is no deterioration measurement available, the rescheduling is performed normally by only considering the new mission and the remaining ones.
- **No new mission is requested but a deterioration measurement is recorded.** For the application examples, we assume that this measurement is available at the end of some of the missions among the mission list.
 - ◆ If the mission is at the end of a block in the schedule, we estimate the possible gains generated by a rescheduling with respect to the current schedule.
 - ▶ If they are greater than a minimum rescheduling cost gain ΔC_{min} , a rescheduling is applied.
 - ▶ Otherwise, the current schedule is kept.
 - ◆ When the mission is inside a block, the first question is to know if the block can be completed or if the failure risk exceeds the maximum admissible failure probability \mathbb{P}_{max} . We then estimate the remaining useful lifetime (RUL) for the block using the deterioration measurement and an equivalent Gamma process based on the parameters of the remaining missions to complete to finish the block.
 - ▶ If it is not possible to finish the block without taking too many risks, a rescheduling is performed.
 - ▶ If the block can be completed with a small enough risk, we estimate the possible gains driven by a rescheduling. If they are greater than a minimum rescheduling cost gain ΔC_{min} , a rescheduling is applied. Otherwise, the current schedule is kept. ΔC_{min} plays an important role in the decision-making process. Indeed, it is the minimum cost gain a new schedule has to generate to replace the current schedule. It enables to decide whether a rescheduling is cost-effective or not. It is only used when deterioration information is available as failure occurrences and new missions make rescheduling mandatory. It

acts as a leverage to avoid having too many rescheduling occurrences if the monitoring information does not have a significant cost-based impact on the current schedule.

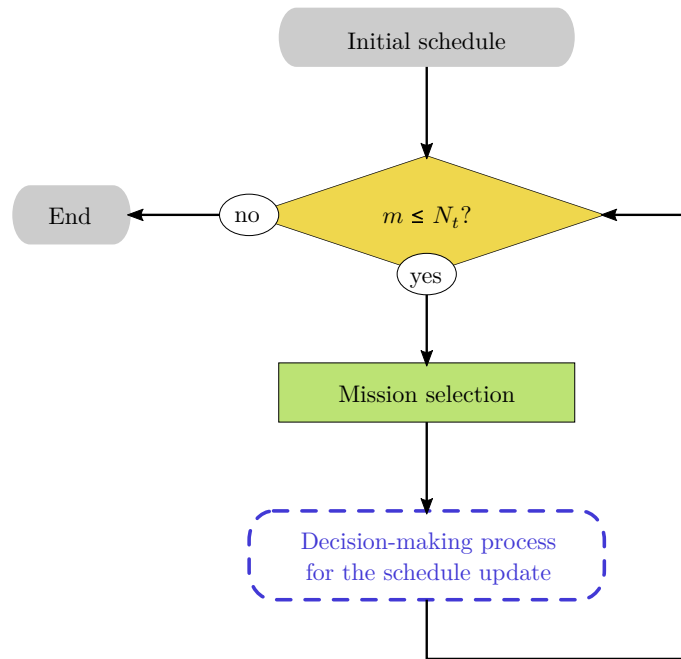


Figure 6.2: Dynamic sequential algorithm principle

The described decision strategy is represented in [Figure 6.3](#). The way to fill in the blocks can slightly vary according to the available information we consider. Indeed, a failure occurrence, caused by an excess of deterioration D with respect to the failure threshold L , implies that the first mission to plan when defining the new schedule is the remaining part of the current mission during which the failure happened. When the deterioration level D is available, it adds a condition on the way to compute the remaining useful lifetime for the first block of the new schedule. The new failure threshold to consider becomes $L - D$. It enables to be more accurate when filling the first block with missions.

The scheduling method is a predictive-reactive strategy insofar as we start from an initial schedule and update it according to some occurring events representing by available monitoring information or new missions availability. Sequential rescheduling enables to adapt at best the schedule to both the real vehicle deterioration progress and the evolution of the delivery activity with new missions to complete.

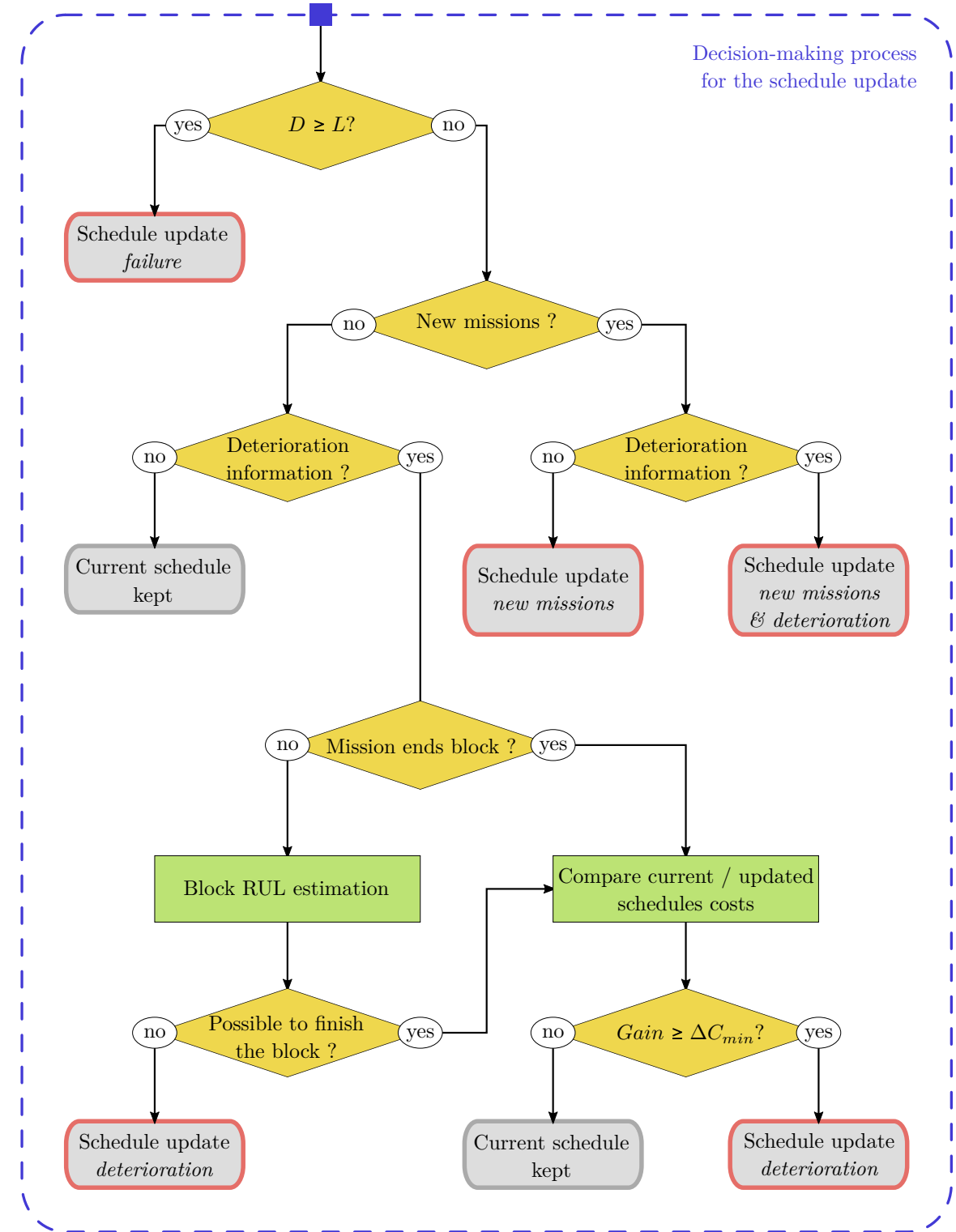


Figure 6.3: Decision-making process for the schedule update

6.2.3.2 Changes in the genetic algorithm functions

The dynamic scheduling implementation is based on the genetic algorithm developed to solve the static scheduling problem. However, some slight changes in the different stages of the genetic algorithm are introduced to consider the available information.

Considering a failure occurrence

After a failure, we know that the vehicle goes to a workshop to be repaired. It means that its deterioration level is back to 0. The first mission to be scheduled in the updated schedule is the remaining part of the one during which the failure occurred. Every schedule generated at each stage of the genetic algorithm has to start by this mission. This constraint is considered for the population initialization, the crossover, the mutation and the population dispersion stages.

Considering a deterioration measurement

An available deterioration measurement enables to estimate more accurately the failure probability of the first block for the updated schedule. This change is integrated in every stage of the genetic algorithm for which schedule are generated i.e. the population initialization, the crossover, the mutation and the population dispersion stages.

Integrating new mission(s)

We have to define which missions are still available when we reschedule and only consider these ones to define a feasible schedule. If the rescheduling occurs at the end of a mission inside a block, the block is not yet finished. The first block of the new schedule is then an extension of the already started block. The failure probability estimation for the first block of this new schedule is assumed to be only based on the missions inside the block. The previous missions completed during the already started block are then not considered. Note that if at the same time a deterioration measurement is available, the problem raised just above is solved.

Based on these comments, we then have four versions of the initial genetic algorithm adapted to the real-time information available. It is either to consider:

1. a failure occurrence
2. a deterioration measurement
3. a new required mission
4. both a deterioration mission and a new required mission.

6.3 Evaluation of the dynamic scheduling method

This part is dedicated to the analysis of the dynamic scheduling algorithm behaviour and compares the scheduling results with the ones obtained for the static scheduling method. This analysis is composed of two parts. The first one aims at showing the interest of integrating the failure occurrences and the deterioration information to update sequentially the schedule while the second one presents the same analysis with the same rescheduling opportunities but with adding a new one based on the availability of new missions.

The analysis is carried on through different application examples to estimate the potential gains generated by considering the real time information available thanks to the monitoring systems and the customer activity.

6.3.1 Rescheduling opportunities: failure occurrences and deterioration information

This section draws a comparison between the dynamic scheduling method when considering the failures and the deterioration measurements and the static scheduling method [142]. This comparison is managed through analyses on the maintenance cost convergence, even for larger size problems and the rescheduling condition effect.

Two datasets with $n = 6$ missions are considered: the dataset A defined in Table 6.1 and the dataset derived from dataset A for which the variance of the deterioration is increased by a factor 2 (Table 6.2). This variance change is studied to see its impact on the optimal schedule.

The dataset C (Table 6.3) with $n = 20$ missions is also considered to compare the algorithms for larger-size problems.

Table 6.1: Dataset A

Missions	Durations(h)	α_m	β_m	Failure probabilities
1	21	0.13	0.1	0.002
2	21	0.18	0.1	0.009
3	8	0.40	0.1	0.004
4	8	0.33	0.1	0.002
5	2	1.33	0.1	0.002
6	3	1.32	0.1	0.01

The parameters for both the genetic algorithm, the maintenance costs and the failure threshold are the same as the one defined for the static case application example (see Table 5.5). The maximum admissible failure probability conditioning the block filling in the algorithms is fixed at $\mathbb{P}_{max} = 0.1$.

Table 6.2: Datasets when increasing the variance by 2 or 5 and when decreasing the variance by 2

Missions	Variance x2			Variance x5			Variance /2		
	α_m	β_m	\mathbb{P}	α_m	β_m	\mathbb{P}	α_m	β_m	\mathbb{P}
1	0.07	0.05	0.02	0.03	0.02	0.05	0.27	0.2	4.2×10^{-5}
2	0.09	0.05	0.04	0.04	0.02	0.09	0.37	0.2	5.8×10^{-4}
3	0.20	0.05	0.02	0.08	0.02	0.07	0.79	0.2	1.1×10^{-4}
4	0.16	0.05	0.01	0.07	0.02	0.05	0.66	0.2	2.5×10^{-5}
5	0.66	0.05	0.01	0.27	0.02	0.05	2.65	0.2	2.7×10^{-5}
6	0.66	0.05	0.04	0.26	0.02	0.09	2.64	0.2	7.2×10^{-4}

After specifying the parameters, the last point to discuss is the choice of the decision criterion for each algorithm.

6.3.1.1 Chosen decision criterion for the algorithms

As explained previously, two examples are studied to see the interest of integrating monitoring information to update the initial schedule. In this section, we study the integration of failure occurrences and deterioration measurements to update the schedule. In this study, we assume that the starting deadline for the missions is not an available feature to define the mission priority. They all have the same priority level.

We can then choose to optimize only the global maintenance cost as the starting deadlines are not available. Indeed, starting deadlines have an major interest when considering the production cost. For the dynamic algorithm, as a rescheduling occurs every time there is a failure, considering multi failure probabilities to estimate the global maintenance cost may not be necessary. We then adopt the criterion C_1 (Eq. 5.10) as decision criterion for the dynamic algorithm.

For the static algorithm, the obtained schedule is never updated no matter what happens during the schedule completion. In this case, it may be relevant to consider the possibility to have more than one potential failure by block of missions to avoid taking unnecessary risks. We then favour the criterion C_3 (Eq. 6.1) as decision criterion for the static algorithm.

As we only consider the maintenance costs and not the starting deadlines of the missions, the duration of the different maintenance operations is not considered in Section 6.3.1.

6.3.1.2 Cost convergence analysis

A comparison between the maintenance costs generated by the dynamic and static schedules is drawn to study the potential cost gains using Monte-Carlo simulations. The simulation framework is defined as follows.

Table 6.3: Dataset C

Missions	Durations (h)	α_m	β_m	Failure probabilities
1	19	0.20	0.1	0.009
2	11	0.30	0.1	0.004
3	6	0.61	0.1	0.007
4	23	0.14	0.1	0.004
5	7	0.36	0.1	0.001
6	4	0.92	0.1	0.007
7	18	0.15	0.1	0.002
8	3	1.20	0.1	0.006
9	10	0.33	0.1	0.005
10	6	0.41	0.1	0.001
11	5	0.56	0.1	0.002
12	4	0.63	0.1	0.001
13	25	0.11	0.1	0.002
14	9	0.40	0.1	0.007
15	18	0.18	0.1	0.004
16	14	0.25	0.1	0.005
17	22	0.14	0.1	0.003
18	9	0.42	0.1	0.008
19	5	0.79	0.1	0.01
20	13	0.24	0.1	0.003

- A deterioration measurement is available at the end of each mission.
- Deterioration trajectories are generated for the missions. We have 1000 different trajectories for each mission so 1000 simulation scenarios are defined.
- For the static method: for each simulation scenario, the algorithm is applied to obtain the optimal schedule. Then, this schedule is executed.
- For the dynamic method: for each simulation scenario, the algorithm is applied to obtain the initial schedule. Then, this schedule starts being executed. When a real-time information is available, the schedule can be updated and the schedule execution goes on.

The simulations enable to record the real maintenance cost for each method due to preventive and corrective maintenance operations occurring during the schedule completion.

In this part, we set the minimum rescheduling cost gain ΔC_{min} at $\frac{C_0}{2}$ to condition the potential rescheduling occurrences when deterioration information is available. This value corresponds to half the cost of a preventive maintenance operation.

For the simulations, the schedules obtained with the static method for the dataset A

and the dataset variance x2 are respectively π_{s_A} and $\pi_{s_{x2}}$, detailed in Eq. 6.8. The only differences come from permutations of the blocks composing the schedules. But, it has no impact on the obtained maintenance cost through the simulations. For the dynamic method, the initial schedules are the same as the ones obtained with the static method for both datasets respectively.

$$\pi_{s_A} = \{ [2][3,4][1,5][6] \} \quad \text{and} \quad \pi_{s_{x2}} = \{ [5,4][6][1][3][2] \} \quad (6.8)$$

Figure 6.4 shows the maintenance cost convergence through the 1000 simulation scenarios for the datasets A and variance x2. It is a comparison between the static and dynamic methods. We observe that the dynamic method enables to make maintenance cost savings with respect to the static method. The savings for the datasets A and variance x2 respectively represent 15.4% and 25.2% of the maintenance cost.

We note that when the variance of the missions deterioration processes increases, the savings are higher. For high variance values, the monitoring information carries more information on the vehicle behaviour with respect to the average vehicle behaviour. A higher variance means also a higher risk to have a failure during the mission completion. The schedule defined with the static method is then composed of more blocks to avoid the failures. However, as the monitoring information on the vehicle deterioration is not considered to update the schedule, there is no chance to reduce the initial number of blocks composing the schedule.

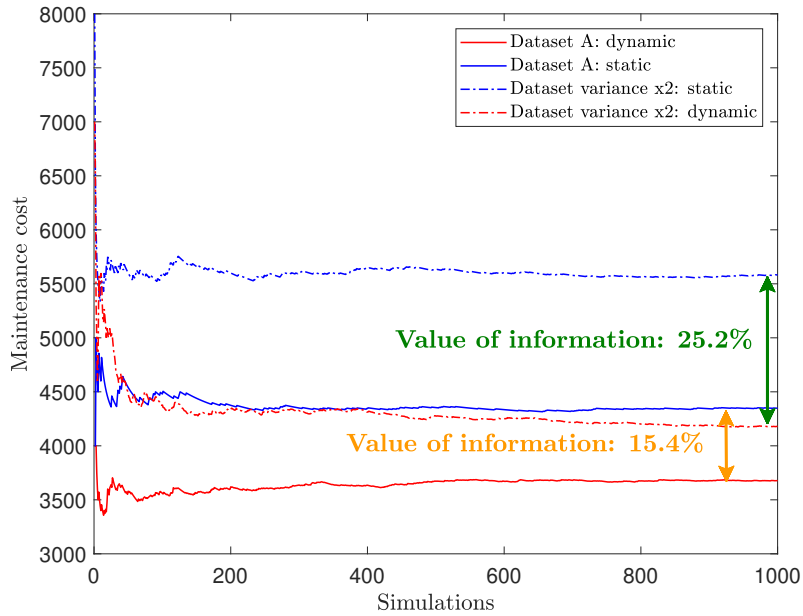


Figure 6.4: Comparison of the maintenance cost convergence through the 1000 simulations between the static and the dynamic methods for the datasets A and variance x2

With the dynamic method, as we can consider the available deterioration measurements, the schedule can change and missions that were apart in the initial schedule could be grouped.

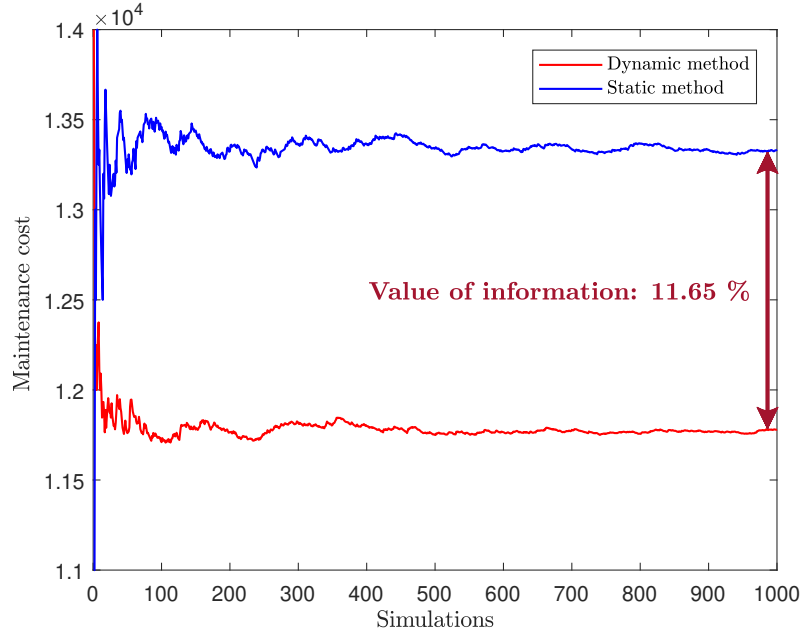


Figure 6.5: Comparison of the maintenance cost convergence through the 1000 simulations between the static and the dynamic methods for dataset C

This could reduce the number of blocks, thus reducing the number of preventive maintenance operations, thus saving money. This is all the more true when the variance of the mission degradation processes is high.

Figure 6.5 shows that the deviation between the maintenance costs for the dynamic and static methods when considering the dataset C (Table 6.3). Indeed, the maintenance costs for the dynamic method converge towards 11781 while they converge towards 13334 for the static method. Using the monitoring information to reschedule enables to save 11.65% of the maintenance costs.

A dynamic maintenance decision based on richer information is then all the more relevant than a static maintenance decision when no monitoring information is considered. This part clearly highlight the importance of considering the monitoring information to make decisions and adapt the initial schedule for missions and maintenance. Monitoring information is really valuable to make the best decisions and avoid unnecessary maintenance expenses.

6.3.1.3 Effect of the rescheduling condition ΔC_{min}

The maintenance costs are influenced by the rescheduling condition depicted by the minimum rescheduling cost gain ΔC_{min} . It is the minimum cost gain a new schedule has to generate to replace the current schedule and enables to decide whether a rescheduling is cost-effective. Using the same simulation framework and similar simulation scenarios as in Section 6.3.1.2,

we study the evolution of the maintenance costs when ΔC_{min} varies for the dataset A. We then compare it with the static schedule maintenance cost.

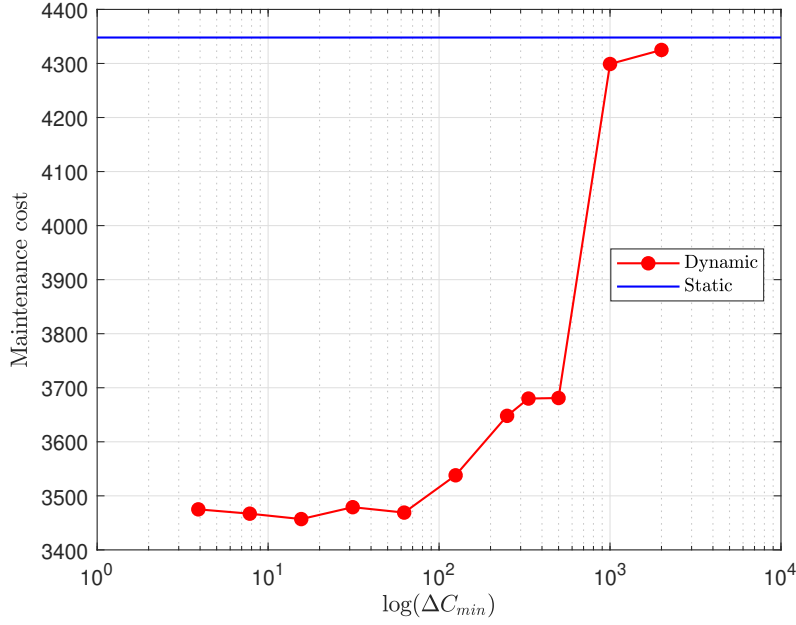


Figure 6.6: Comparison of the maintenance cost convergence between the static and the dynamic methods when ΔC_{min} is varying

Figure 6.6 presents the maintenance costs convergence points for the static and dynamic methods when the minimum rescheduling cost gain ΔC_{min} varies between $2C_0$ and $\frac{C_0}{256}$. We can globally observe that the maintenance costs for the dynamic methods decrease when ΔC_{min} decreases until the rescheduling almost becomes an automatic action at the end of each mission, as a deterioration measurement is available at the end of each mission. For the static method, the maintenance costs stay the same as there is no rescheduling occurrence. For the static method, the maintenance costs converge towards 4348.

When $\Delta C_{min} \geq C_0$, the maintenance costs are quite similar between the two methods. We can then consider that they are equivalent. A sharp drop happens when ΔC_{min} reaches $\frac{C_0}{2}$. Then, the maintenance costs are still decreasing but with a slower speed. Once ΔC_{min} reaches $\frac{C_0}{16}$, the variations of the maintenance costs are small enough to consider that the maintenance costs are stabilizing.

The maintenance cost values when applying the dynamic method for the different values of ΔC_{min} are displayed in Table 6.4. For each value of ΔC_{min} , the gain earned when using the dynamic method with respect to the static method is computed.

The main costs gains show that encouraging rescheduling more often enables to minimize the maintenance costs. However, in this study, we do not consider the rescheduling costs. Indeed, rescheduling means changing the schedule, so changing the missions order. It could lead to additional logistic costs. By taking it into consideration, an optimal value of ΔC_{min}

Table 6.4: Maintenance costs and gains generated by the dynamic method with respect to the static method according to ΔC_{min}

ΔC_{min}	Maintenance costs	Cost gain (%)
$2C_0$	4325	0.53
C_0	4299	1.22
$\frac{C_0}{2}$	3681	15.34
$\frac{C_0}{3}$	3680	15.36
$\frac{C_0}{4}$	3648	16.10
$\frac{C_0}{8}$	3538	18.63
$\frac{C_0}{16}$	3469	20.22
$\frac{C_0}{32}$	3479	19.99
$\frac{C_0}{64}$	3457	20.49
$\frac{C_0}{128}$	3467	20.26
$\frac{C_0}{256}$	3475	20.08

could be obtained to reach a balance between the rescheduling costs and the gains. With this optimal value, the global maintenance cost gain could be compared with the expenses necessary to retrieve the monitoring information used for the rescheduling occurrences.

6.3.2 Rescheduling opportunities: failure occurrences, deterioration information and new available missions

This section analyses the dynamic sequential algorithm behaviour according to the events that could trigger a schedule update. As for [Section 6.3.1](#), the failure occurrences and the deterioration measurements are considered but the possibility to integrate new additional missions to complete is also taken into account. These missions are requested during the schedule completion so they are unknown at the beginning of the scheduling process and they have to be dynamically accounted for.

The initial mission list is composed of 6 missions (grey part of [Table 6.5](#)). The next 12 missions to complete are progressively added to the mission pool and have to be integrated to the on-going schedule. [Table 6.5](#) describes the features of the 18 missions. The missions 1 to 6 compose the initial mission list.

Table 6.5: Dataset D

Mission	Duration (h)	α_m	β_m	Failure probabilities	Starting deadlines
1	21	0.1335	0.1	0.0021	90
2	21	0.1837	0.1	0.0087	12
3	8	0.3959	0.1	0.0035	32
4	8	0.3285	0.1	0.0015	2
5	2	1.3254	0.1	0.0016	43
6	3	1.3206	0.1	0.0099	110
7	3	1.0150	0.1	0.0030	45
8	10	0.4216	0.1	0.0132	48
9	13	0.2465	0.1	0.0037	70
10	44	0.1041	0.1	0.0195	200
11	19	0.1937	0.1	0.0070	150
12	9	0.4043	0.1	0.0067	98
13	13	0.2077	0.1	0.0017	190
14	3	0.8863	0.1	0.0016	160
15	5	0.9130	0.1	0.0192	165
16	3	0.8177	0.1	0.0012	128
17	22	0.1303	0.1	0.0023	165
18	6	0.6972	0.1	0.0127	145

The raw gain g_m is equal to 5000 and the unitary delay cost c_{ud} is equal to 50. We assume that all the missions have the same raw gain g_m and the same unitary delay cost c_{ud} . The other parameters for the genetic algorithm and the maintenance management are the same as for [Section 6.3.1](#) (see [Table 5.5](#)). The maximum admissible failure probability conditioning the block filling in the algorithms is fixed at $\mathbb{P}_{max} = 0.1$.

- **Static simulation of the missions:**

As the static method never considers rescheduling, the first 6 missions are scheduled and executed. Once, they are all over, the 12 remaining ones are scheduled and completed. It is a way to simulate the fact that the static method does not reschedule even when new missions are requested.

- **Dynamic simulation of the missions:**

The occurring events during the schedule completion are:

- 4 new missions are available at the end of the missions 1, 3 and 5;
- Deterioration measurements are available at the end of the missions 1, 2, 5, 6, 8, 10, 12, 13, 14, 17 and 18.

It means that both new missions and deterioration are available at the end of missions

1 and 5. Failures occur randomly during the schedule completion but once there is one, it triggers a rescheduling.

6.3.2.1 Chosen decision criterion for the algorithms

For both algorithms, we choose to use a criterion based on the operating incomes for the vehicle. It considers both the maintenance costs and the production gains earned when the missions are completed.

The maintenance costs are estimated based on the criterion C_3 considering multi failures probabilities. The production gains are estimated based on C_4 but using a simplification. We consider that the starting deadlines for the missions inside a block can be assimilated to the date when the block starts. The operating incomes correspond to the difference between the production gains and the maintenance costs. The chosen criterion C_5 for a schedule π is then defined in Eq. 6.9.

$$C_5(\pi) = \sum_{i=1}^n \left(g_m(i) - c_{ud}(i) \times \max \left\{ 0; \mathbb{E} \left[t_s(\{k \mid i \in k\}) \right] - d_{s_{max}}(i) \right\} \right) - \sum_{k=1}^{N_b} \left(C_0 + C_f \sum_{i=1}^{N(k)} \mathbb{P}_f(k, i) \right) \quad (6.9)$$

To estimate the starting deadlines of the missions, we consider the durations of the maintenance operations. These durations are:

- $d_p = 2$ for the preventive maintenance operation duration,
- $d_c = 4$ for the corrective maintenance operation duration.

6.3.2.2 Cost convergence analysis

This section compares the results between the dynamic and the static methods in terms of operating incomes, computation time and final executed schedule. To compare these results, Monte-Carlo simulations are generated as for Section 6.3.1. The only difference is that we consider only 250 simulation scenarios, but it is sufficient to reach the operating incomes convergence.

The simulation framework is defined as follows:

- Deterioration trajectories are generated for the missions. We have 250 different trajectories for each mission so 250 simulation scenarios are defined.
- For the static method: for each simulation scenario, the algorithm is applied to obtain the optimal schedule for the 6 initial missions. Then, this schedule is executed. Once the missions are completed, the 12 others are scheduled and completed.

- For the dynamic method: for each simulation scenario, the algorithm is applied to obtain the initial schedule with the 6 missions. Then, this schedule starts being executed. When a failure occurs or new missions are available, the schedule is updated. If a deterioration measurement is available, the schedule can be updated. After each rescheduling, the schedule execution goes on.

The initial schedule for both methods is similar and is defined by $\pi_{is} = \{ [4, 2] [3] [5, 1] [6] \}$. However, the final schedule is different insofar as the dynamic method reschedules the remaining missions according to the available monitoring information and the new missions requests.

For each method, Monte-Carlo simulations enables to compare the operating incomes generated by the schedules and the computation time necessary to obtain them. Let π_{ds} (Eq. 6.10) and π_s (Eq. 6.11) be examples of the final schedules respectively for the dynamic and static methods. The schedule π_{ds} only has 9 blocks while π_s has 13 blocks. So, π_s performs four more maintenance operations. It explains why the operating income for the dynamic method is equal to 76370 while the one for the static method is 70182. In addition, π_{ds} contains less blocks of one mission than π_s . Then, the missions are more grouped into blocks with the dynamic method.

$$\pi_{ds} = \{ [4, 2, 5] [3, 8, 7] [6] [9, 1, 16] [12, 18] [11, 14] [15] [13, 17] [10] \} \quad (6.10)$$

$$\pi_s = \{ [4, 2] [3] [5, 1] [6] [7] [8] [9, 16] [15] [12] [14, 13] [18] [17, 11] [10] \} \quad (6.11)$$

Figure 6.7 shows that the dynamic method is more profitable than the static method by about 8.82%. The benefit represents more than the cost of two corrective maintenance operations. Note that the dynamic method is more time-consuming than the static one. Indeed, the static method requires around 10 seconds to compute and execute the schedule while the dynamic method needs about 50 seconds. The dynamic method is then five times longer.

Figure 6.8 depicts the distribution of the total number of rescheduling n_t according to the Monte-Carlo simulations made with the dynamic method. On average, five schedule updates are performed to schedule and complete the 18 missions. For each simulation, the number of rescheduling can be divided into four parts according to the events inducing the updates:

- n_f : number of rescheduling induced by failure occurrences,
- n_d : number of rescheduling caused by deterioration measurements,
- n_m : number of rescheduling caused by requests of new missions,
- n_{md} : the number of rescheduling induced by both new missions requests and deterioration measurements.

For each cause, the rescheduling strategy differs. The total rescheduling number n_t is then defined by Eq. 6.12.

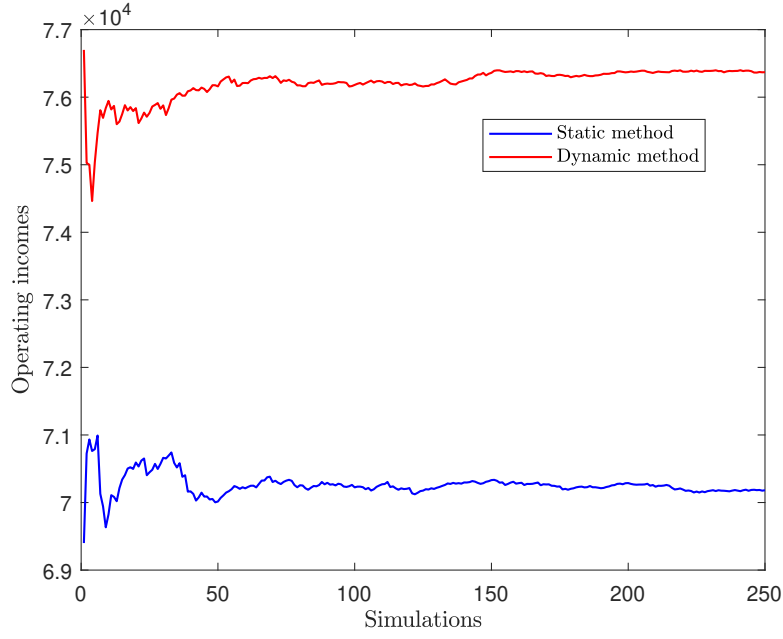


Figure 6.7: Comparison of the operating incomes between the static and the dynamic methods for $\Delta C_{min} = \frac{C_0}{2}$

$$n_t = n_f + n_d + n_m + n_{md} \quad (6.12)$$

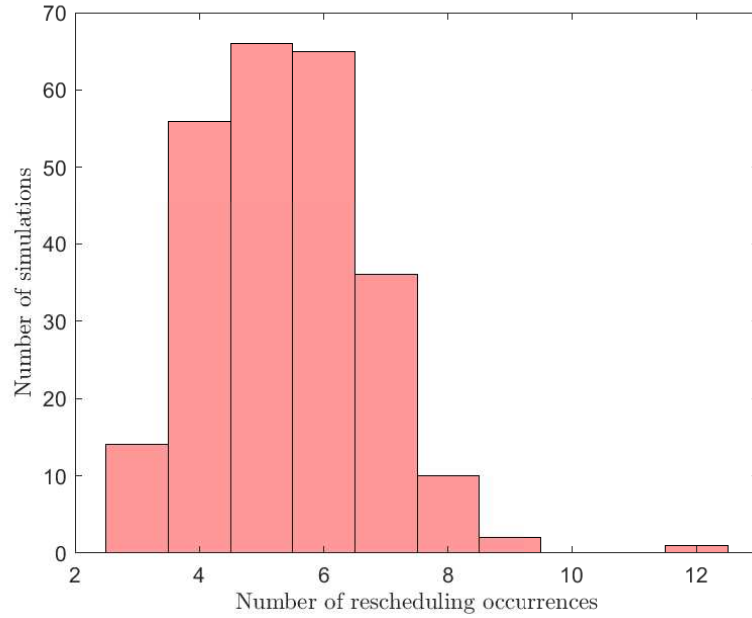


Figure 6.8: Distribution of the total number of rescheduling n_t

Figure 6.9 illustrates the split of the total number of rescheduling according to their causes.

Based on the simulation framework, two schedule updates always happen. It is due to new requested missions and deterioration information available at the end of the missions 1 and 5. Indeed, updates when new missions are available is mandatory. Likewise, one update also always occurs as new missions are requested at the end of mission 3. It explains why the 250 simulations have $n_{md} = 2$ and $n_m = 1$ (cf the two graphs at the bottom of Figure 6.9).

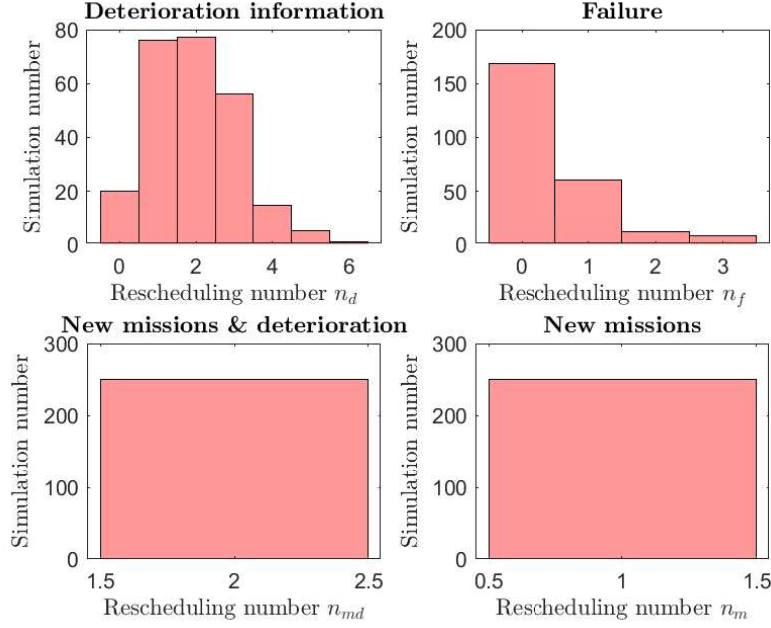


Figure 6.9: Distribution of the numbers of rescheduling by rescheduling cause (n_d , n_f , n_{md} and n_m)

The gains earned with the dynamic method comes from the schedule updates applied when failures occur or new missions are requested, or when information on the vehicle health state is gathered. We consider that failures and new missions inevitably lead to a rescheduling of the current schedule. In fact, failures imply that the on-going schedule was not well enough designed for the vehicle usage. This circumstance also allows us to know that after the corrective maintenance operation performed to repair the failure, the vehicle deterioration level is back to 0. This piece of information is useful to schedule the remaining missions accordingly. That is why, rescheduling after a failure is mandatory. Likewise, when new missions arrive in the mission pool, the schedule must be updated to integrate them as soon as possible. It can lead to avoid potential delay costs due to the missions starting deadlines characterizing their priority. For instance, if a new top priority mission is demanded, it will be placed in the schedule to maximize the criterion value. A balance between the estimated gains, delay costs and maintenance costs have then to be reached. Still, rescheduling in this condition is also mandatory to prevent possible delay costs when the new missions will be completed.

6.3.2.3 Effect of the rescheduling limit condition ΔC_{min}

When it comes to rescheduling, the only flexibility lies in taking into account the vehicle health state information. Indeed, in this case, a rescheduling is not mandatory. That is why, the minimum rescheduling cost gain ΔC_{min} is introduced as a rescheduling limit condition. It enables to decide whether a rescheduling is cost-effective. If the cost difference between the current schedule and the new one does not exceed ΔC_{min} , a rescheduling is not applied and the current schedule remains as it is.

Figure 6.10 presents a comparison on the operating incomes obtained between the static and dynamic methods for different values of ΔC_{min} . We assume that no additional cost are induced when a rescheduling occurs. We can observe that the operating incomes increase when ΔC_{min} decreases.

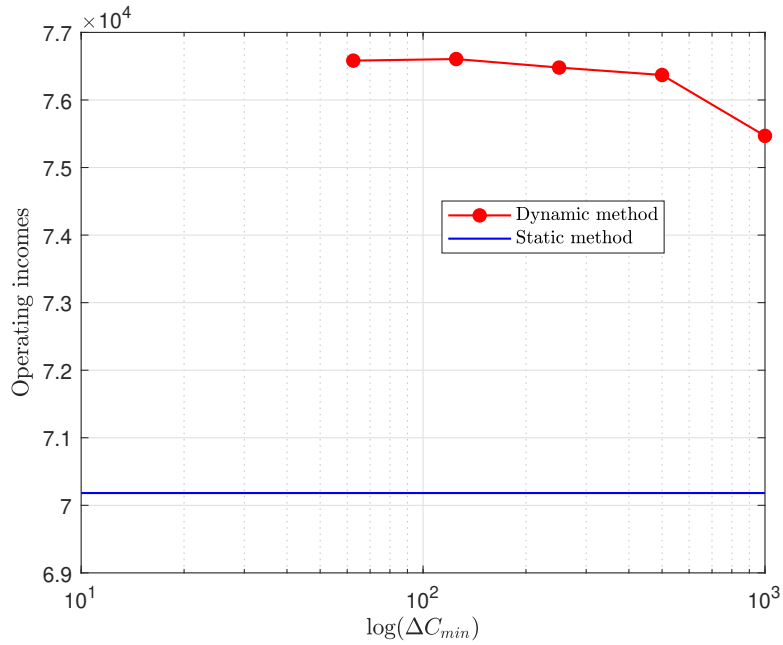


Figure 6.10: Comparison of the operating incomes between the static and dynamic methods when ΔC_{min} is varying

Table 6.6 contains all the operating income values, the income gains generated with respect to the static method and the average number of rescheduling according to the values of ΔC_{min} that limits the rescheduling when deterioration measurements are available. The operating incomes for the static method are always equal to 70182. However, the difference of operating incomes between the different values is not as significant as when only the failures and deterioration measurements are considered.

Table 6.7 gives the average number of rescheduling on all the simulations for each of the four rescheduling causes. As expected, as rescheduling is mandatory when new missions are available, n_{md} and n_m are always respectively equal to 2 and 1 due to the new missions added

at the end of missions 1, 3 and 5. The deterioration measurements available when missions 1 and 5 are over is an added bonus for the rescheduling accuracy. The average values for n_f let us assume that there are not many failures occurring during the schedule completion. It is good news as the objective is to avoid at best failure occurrences. The number of rescheduling due to deterioration measurements increases as ΔC_{min} decreases. It is natural because we allow more schedule updates to be applied.

Table 6.6: Operating incomes and gains generated by the dynamic method with respect to the static method according to ΔC_{min}

ΔC_{min}	Operating incomes	Cost gain (%)	Average number of rescheduling
C_0	75469	7.53	4.34
$\frac{C_0}{2}$	76370	8.82	5.39
$\frac{C_0}{4}$	76480	8.97	5.96
$\frac{C_0}{8}$	76606	9.15	6.22
$\frac{C_0}{16}$	76582	9.12	6.57

Table 6.7: Average value of the number of rescheduling according to their associated cause and the value of ΔC_{min}

ΔC_{min}	$\overline{n_d}$	$\overline{n_f}$	$\overline{n_{md}}$	$\overline{n_m}$
C_0	0.84	0.50	2	1
$\frac{C_0}{2}$	1.96	0.44	2	1
$\frac{C_0}{4}$	2.48	0.48	2	1
$\frac{C_0}{8}$	2.72	0.50	2	1
$\frac{C_0}{16}$	3.05	0.52	2	1

Choosing the right value for ΔC_{min} is essential to find a balance between the potential gain offered by the monitoring information and the costs induced by the monitoring systems set up and the monitoring data collection. We do not consider the logistic costs that could be related to a rescheduling. However, as the starting deadline is a feature characterizing the missions, integrating the delay costs estimation when optimizing the schedule is an alternative way to consider the additional costs and limit the disruption associated with a rescheduling.

6.3.3 Rescheduling analysis for different criteria related to the operating incomes estimation

When discussing the different possibilities to define a decision criterion in [Section 6.2.2](#), we explained that estimating the starting dates for the missions was the hardest part and that some simplifications could be pondered.

Based on the simulation framework and on the study case defined in [Section 6.3.2](#), we analyse in this section the convergence of the operating incomes and of the computation time based on three different criteria.

For all the criteria estimating the operating incomes, the global maintenance costs estimation is based on the criterion C_3 (Eq. 6.1). The only changing part is on the production costs estimation and in particular on the way to estimate the missions starting dates.

The three criteria are:

1. The criterion C_5 (Eq. 6.9) that assimilates the estimated starting dates of the missions inside a block to the estimated date when this block begins.
2. The criterion C_6 that considers to estimate the starting date of a mission i in the block k both the durations of the missions set before mission i in the block k and the estimated date when block k starts. The time spent in block k before executing mission i defined in Eq. 6.6 is then reduced to the sum of the durations for missions before mission i .
3. The criterion C_7 (Eq. 6.13) is based on the criterion C_4 to estimate the production costs. The starting date for mission i in block k is based on the estimated date when block k starts, on the missions durations for the missions before i in block k and the average number of failures that could occur during these previous missions in block k .

$$C_7(\pi) = \sum_{i=1}^n \left(g_m(i) - c_{ud}(i) \times \max \left\{ 0; \mathbb{E} \left[t_s(\{k \mid i \in k\}) \right] + t(\{k \mid i \in k\}, i) - d_{max}(i) \right\} \right) - \sum_{k=1}^{N_b} \left(C_0 + C_f \sum_{i=1}^{N(k)} \mathbb{P}_f(k, i) \right) \quad (6.13)$$

Operating incomes convergence estimation

[Figure 6.11](#) studies the operating incomes convergence when considering the different criteria C_5 , C_6 and C_7 to optimize the schedule obtained with the dynamic method. The operating incomes converge towards 76523, 77562 and 77470 respectively when using C_5 , C_6 and C_7 . The difference of incomes with C_6 and C_7 with respect to C_5 ranges from 1.25% to 1.36%. It stays small but the income difference is related to the way to estimate the starting deadlines for the missions in the criteria. Indeed, it can lead to a different decision-making process to define the schedule but also when it comes to decide whether to reschedule or not when deterioration information is available.

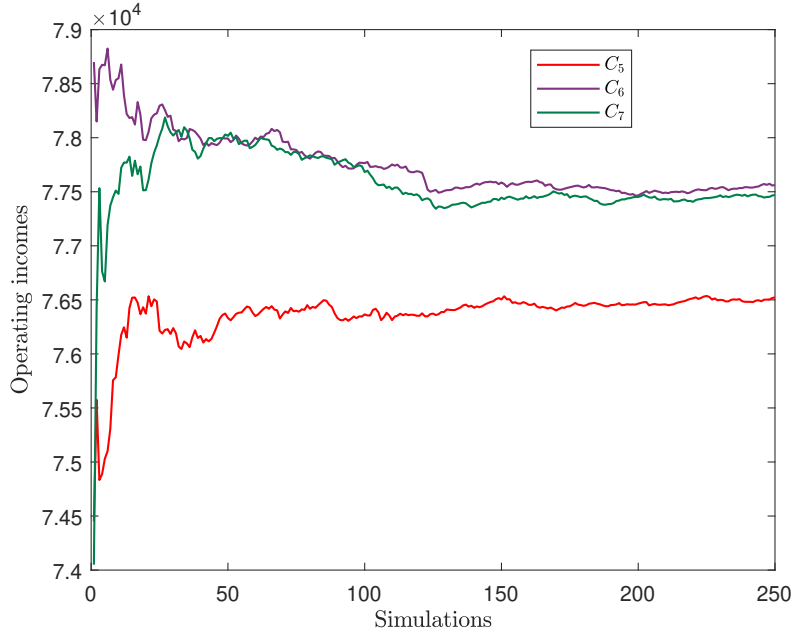


Figure 6.11: Comparison of the operating incomes between the three different criteria for the dynamic method

Computation time convergence analysis

Figure 6.12 displays the convergence of the computation time when using C_5 , C_6 and C_7 as optimization criteria. As expected, when considering C_5 , there are less calculations to do to estimate the starting date for the missions. Indeed, the missions inside the same block are assumed to start when the block starts. We then take more risk to have delay costs. The criterion C_7 is the one leading to the highest computation time insofar as, for each mission i belonging to a block k , it is necessary to estimate the times spent in the block k before starting the mission i . It includes the durations of all the missions in block k before mission i (like for C_6) but also an estimation of the average number of failures that could happen during these missions to estimate the potential time spent in maintenance.

The convergence computation time values for C_5 , C_6 and C_7 are respectively equal to 41.06s, 44.48s and 45.51s. The difference seems small but if the number of rescheduling and the number of missions to schedule at the same time increase, these time difference could drastically increase.

If we want to have a close estimation of the expected operating incomes, it is better to use the criterion C_7 . The criterion C_6 can be also a good alternative to have a good estimation of the operating incomes, so a good optimized schedule, without increasing too much the computation time to keep the reactivity of the algorithm when a schedule update is needed. If we want to favour the algorithm reactivity when rescheduling, the criterion C_5 could be more adapted. Choosing the right decision criterion is always a question of balance between the optimal solution and the computation time. The computation time may not be seen as

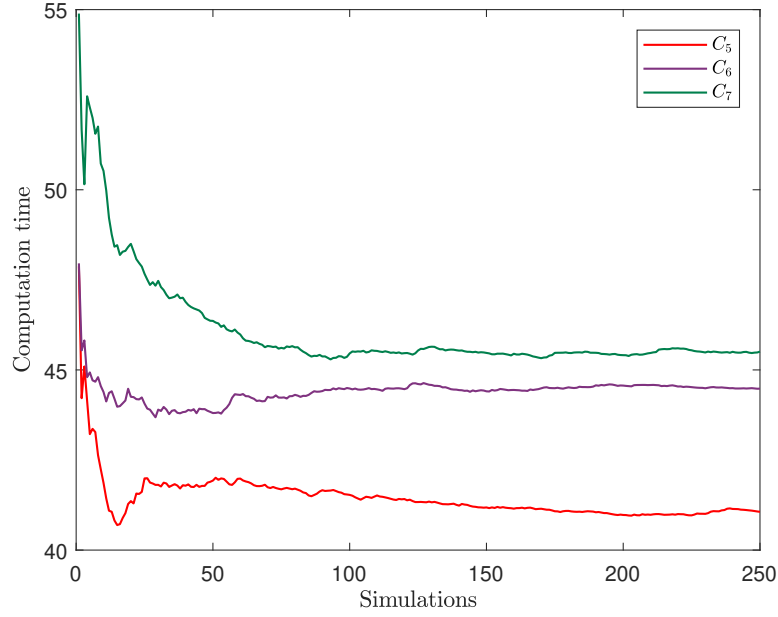


Figure 6.12: Comparison of the computation time between the three different criteria for the dynamic method

a problem but if this rescheduling methodology is implemented through an app and given to the customer to manage his fleet of vehicles, he is expecting the algorithm to give him an optimal schedule as fast as possible so he can make the necessary logistic arrangements to perform the new mission order.

6.3.4 Limits of the dynamic scheduling algorithm

There are two major points that can be seen as limits for the developed algorithm to solve the dynamic joint scheduling problem.

The first one is related the reactivity of the dynamic method. Indeed, if the size of the problem increases, the mission pool becomes bigger and the genetic algorithm parameters need to be tuned accordingly to ensure the algorithm convergence towards a satisfying schedule. This point can lead to a much higher computation time to obtain a schedule at a certain moment. So, it could reduce the algorithm reactivity. The same comment can be made if we do not often reschedule but many missions arrive at the same time.

The second point deals with the management of the rescheduling opportunities. Indeed, if we often reschedule, it means that the current schedule is not stable enough and the schedule is then not trustworthy on the long-term [170]. Naturally, the rescheduling frequency depends a lot on the frequency to collect monitoring data and to request new missions. But it also depends on the mission priority distribution. If all the missions have the same priority (start-

ing deadline), rescheduling can completely change the current missions order for execution without much significance or necessity. It will only lead to additional logistic costs to organize the products to deliver. The introduction of ΔC_{min} is a first step to handle the rescheduling frequency. In addition, if there are many rescheduling, it also leads to a higher computation time.

6.4 Conclusion

The dynamic joint scheduling problem has been defined in this chapter and an adapted genetic algorithm based method has been developed to initiate a schedule and update it according to the monitoring data that can be collected and the production activity evolution due to new requested missions. Failure occurrences and new missions requests are considered as mandatory opportunities to reschedule the remaining activities to be sure to fit to the vehicle usage and avoid delay costs by not scheduling soon enough high-priority missions. Deterioration measurements regarding the vehicle health state are also retrieved and can be used to update the schedule if the current block of mission cannot be finished without a too high failure risk. If it is not the case, a rescheduling is applied only if this update enables to make some significant cost savings.

We observe through the different application examples that considering real-time events as opportunities to reschedule leads to significant maintenance cost savings and enables to increase the operating income for the vehicle.

Rescheduling clearly improves the operating incomes but we need to be careful regarding the rescheduling management. In fact, rescheduling too often can create an instability of the schedule, which means that we cannot rely on the schedule for a long time period because it presents little predictability. It really is a question of balance between the number of rescheduling and the criticality of the information used to trigger schedule updates. In addition, we do not consider the spatial position of the trucks in our study. It would be relevant to take it into account in a further study to optimize the rescheduling opportunities also based on the location of the trucks according to the delivery points for the missions and the workshops location.

Volvo customers have very different profiles. Some only have one vehicle and try to optimize their activity. In this case, the dynamic method developed for one single vehicle could be useful to guide them on how to schedule both their missions and their maintenance operations. But others have an entire fleet of vehicle to manage. When they have some missions to perform, it is necessary to choose the right vehicle for the right missions based on the monitoring information, the vehicles availability and on the vehicles configuration. The following chapter treats the joint scheduling problem for a whole fleet of vehicles to dispatch the right vehicle on the right mission according their health state and their availability as well as the usage conditions of the requested missions.

Joint scheduling of missions and maintenance operations for a fleet of vehicles

Contents

7.1	Fleet management problem	164
7.2	Problem description	165
7.2.1	Hypotheses and constraints	165
7.2.2	Objectives	166
7.3	Reminder: vehicle deterioration model	167
7.4	The static case	168
7.4.1	Resolution approach description	169
7.4.1.1	Decision criterion	169
7.4.1.2	Description of the GA difference between one vehicle and a fleet	170
7.4.2	Performance of the fleet static scheduling approach	171
7.4.2.1	Example for an homogeneous fleet of vehicles	172
7.4.2.2	Example for an heterogeneous fleet of vehicles	177
7.5	The dynamic case	181
7.5.1	Resolution approach description	181
7.5.1.1	Definition of the missions features	181
7.5.1.2	Decision criterion	182
7.5.1.3	Description of the rescheduling method for the fleet	184
7.5.2	Application example to analyse the performances of the fleet dynamic scheduling method	186
7.5.2.1	Analysis of the fleet schedule evolution on a simulated scenario	188
7.5.2.2	Performance analysis	192
7.6	Conclusion	198

In the previous chapters, we study the joint dynamic scheduling problem for a single vehicle and consider the failure occurrences, the monitoring information about the vehicle deterioration and the demand for new missions as opportunities to update the initial schedule. This methodology is satisfactory to schedule the delivery activity and the maintenance

operations to perform according to the vehicle health state evolution and according to the evolution of the requested missions.

However, when the customer owns a fleet of vehicles, he has to make decisions regarding the missions assignment and the maintenance planning for all his vehicles. Considering the state of the whole fleet, and not of just each vehicle independently, could help making better decisions about how to manage the fleet, i.e. how to dispatch the missions between the different vehicles and when to schedule their maintenance operations. We focus in this chapter on the fleet management aspect and how to make decisions at the fleet level to use at best the fleet capacity while optimizing the expenses associated with maintenance and the gains earned when completing the missions.

We propose a methodology to define a joint schedule for missions and maintenance operations for the whole fleet. In a first step, the static case is studied to only consider one additional complexity level with respect to the static joint scheduling problem for a single vehicle. This first step aims at proving that making decisions while considering the whole fleet enables to generate benefits when compared to making decisions independently from one vehicle to another. Then, we add again a new complexity by integrating different real-time events to potentially update the initial schedule. These events are the same as the ones considered in [Chapter 6](#), i.e. the failure occurrences, the deterioration information and the new requested missions. Application examples in both cases illustrate the interest of considering the fleet dimension in the decision-making process and the added value when applying a dynamic method rather than a static one.

7.1 Fleet management problem

The fleet management problem consists in dispatching the missions or jobs between several systems, in our case vehicles, in the best way possible to optimize a global objective function. It is a cooperative approach.

The fleet management problem can be associated with the load-balancing issue on a distributed system. This problem is quite common in cloud computing and for plug electric vehicles PEV. Indeed, in cloud computing, the objective is to assign or reassign the load among the different available resources to maximize the throughput while minimizing the cost and response time and improving performance and resource utilization [69]. It aims at avoiding the situations where some nodes of the network are overloaded while others have little to do. For the PEV load management issue, the objective is to distribute the load i.e. the energy among the electricity grid according to the charging demand from the PEV so they can reach an adequate state of charge [130].

The load management approaches are divided into two clusters [179]. The first one groups the centralized strategies. They require information each resource's goals and constraints to be available to a central planner which makes decisions for all the resources. The second cluster encompasses the decentralized strategies. These approaches remove the requirements

of central coordination and allow each individual node to determine its own actions based only on locally available information.

The joint scheduling of maintenance and missions for a fleet of vehicles deals with load balancing on a distributed system, the fleet, for which each node i.e. each vehicle contributes to the execution of the mission. It is managed by a centralized approach for which a control tower makes decisions for the whole fleet based on the available information on the different trucks inside the fleet.

7.2 Problem description

This section describes the joint scheduling problem for a fleet in both a static and a dynamic case. The basic problem is similar. The only difference is that rescheduling is possible when studying the dynamic case. The available monitoring information to consider are the deterioration measurements, the failure occurrences and the new requested missions.

7.2.1 Hypotheses and constraints

- **Pool of missions**

For the static case, a pool of missions is available at the beginning of the schedule definition and no monitoring information regarding any vehicle of the fleet is available to update the schedule. On the contrary, for the dynamic case, the initial pool of missions can evolve if new missions are requested.

- **Fleet**

A fleet of N_v vehicles is available to perform all the missions in the pool. Each vehicle health state deteriorates over time and their deteriorations are different from one vehicle to another according to the missions they are dispatched on and the usage severities of these specific missions. The deterioration phenomenon is still assumed to be modelled with a Gamma process.

- **Vehicle configurations**

However, a fleet can be composed of vehicles having different configurations, more or less adapted to specific usages. For instance, medium-duty vehicles, used for a urban distribution usage, have a smaller cylinder capacity while long haulage vehicles, used for a regional or international distribution usage, have a bigger one. The components also can differ. That is why, the impact of the same mission on two trucks with a different configuration can change

the deterioration level at the end of the mission. In our studies, we can consider either homogeneous fleet with same configuration vehicles or heterogeneous vehicles with 2 different types of configuration. It then adds some parameters when defining a mission.

- **Missions modelling**

A mission is defined by some classical features: a duration, a raw gain, a unitary delay cost and a deadline. It is then associated with two sets of parameters for the Gamma process describing the deterioration evolution $(\alpha_{m_1}, \beta_{m_1})$ if the vehicle has the type 1 configuration and $(\alpha_{m_2}, \beta_{m_2})$ if it has the type 2 configuration.

In the previous chapter, we always consider that the missions cannot be stopped once they have started. That is why we considered that new missions were added at the end of specific missions. In this study, we assume that it is possible to stop a mission and even to assign its remaining part to another vehicle. But, of course penalty costs are induced if this mission is postponed or dispatch on another vehicle.

7.2.2 Objectives

Our aim is to define an algorithm that enables to schedule simultaneously the missions and the maintenance operations for the whole fleet. The static case enables to develop such a methodology that sets up the basis to develop an adaptive methodology for the dynamic case.

The global schedule for the fleet, denoted Π , is composed of several sub-schedules, one for each vehicle of the fleet. Each sub-schedule π_v is defined with blocks of missions separated by preventive maintenance operations. These maintenance operations are scheduled in an optimal way to take into account the model of the deterioration evolution for each vehicle.

For a fleet of N_v vehicles, Π is composed of N_v sub-schedules as illustrated in Eq.7.1. The first vehicle schedule is composed of three blocks of missions and a maintenance operation is scheduled at the end of each block i.e. after the missions 2, 5 and 1.

$$\Pi = \begin{cases} \pi_1 = [10, 2] [5] [3, 8, 1] \\ \pi_2 = [12, 20, 25] [4, 30] [13, 18, 15] \\ \vdots \\ \pi_{N_v} = [40, 42, 56, 28] [50, 32, 60, 9] [21, 11, 6] \end{cases} \quad (7.1)$$

The developed methods have to define a decision-making process to dispatch the missions to the different vehicles and schedule the maintenance operations accordingly. Note that all the vehicles can be dispatched on all the requested missions. Each vehicle deterioration evolution is still described by a global health indicator and depends on the usage conditions

but also of the vehicle design. If the vehicles have distinct configurations, the impact of the missions on their deterioration will also be different.

For the dynamic joint scheduling problem, a predictive-reactive strategy has to be developed to adjust the schedule to the occurring events such as failure, deterioration measurements and new requests of missions.

In this chapter, we do not start from scratch. We have to adapt the methods that we have already implemented for a single vehicle to integrate the fleet dimension in the decision-making process.

The optimization criterion can either be only based on the maintenance costs or can integrate both the maintenance costs and the gains earned when completing the missions including the penalty costs if the deliveries are late. We chose to consider a criterion based on the estimation of the operating incomes for the fleet. The decisions will then be made to satisfy the objectives of the whole fleet and not the ones of the vehicles independently.

The objective of this chapter focuses on the study of the fleet dimension to make decisions. That is why, we aim at comparing respectively the static scheduling methodology for one vehicle with the static one for a fleet and the dynamic scheduling method for one vehicle with the dynamic one for a fleet.

7.3 Reminder: vehicle deterioration model

As for the previous chapters, the deterioration of each vehicle of the fleet is characterized by a global health indicator. The deterioration gradually increases over time if no maintenance operation is performed and a Gamma process with varying parameters is applied to model the vehicle deterioration evolution in a changing operating environment. The deterioration model is defined at the block level as the schedule is composed of blocks of missions separated by preventive maintenance operations. Each mission is then associated with a pair of shape and scale parameters of the Gamma process modelling the deterioration evolution.

This deterioration depends on the usage conditions of the vehicle, but another variable has to be considered. Indeed, among a fleet, all the vehicles are not of the same range. Some can be more adapted to specific usage conditions than others. For instance, the Renault T truck is more adapted to a long haulage usage than a Renault D narrow truck that is designed to be used for deliveries in cities. The Renault K truck is specifically designed to be used for construction applications i.e. to drive on bumping roads where the road conditions are severe with rocks and holes.

The same mission dispatched on two vehicles from different ranges will then not have the same impact on their deterioration. To take into account this statement, we assume that the mission is characterized by two different pairs of parameters corresponding to the shape and scale parameters of the Gamma process. We assume that there are at the maximum $N_{v_d} = 2$ possible designs for the vehicles in the fleet. The shape and scale parameters for a mission m

are then:

- α_{m_1} and β_{m_1} if the vehicle has the configuration 1;
- α_{m_2} and β_{m_2} if the vehicle has the configuration 2.

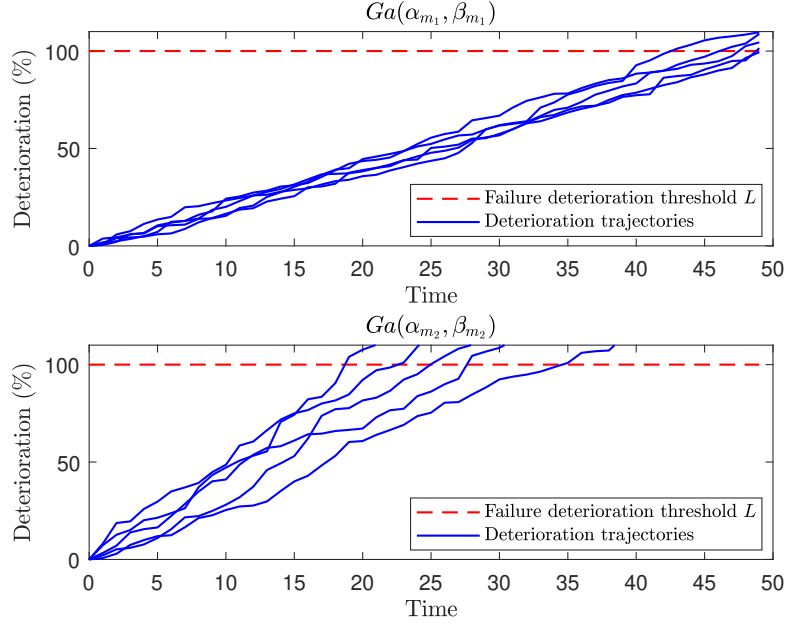


Figure 7.1: Deterioration trajectories for the same mission m for the 2 different configurations

Figure 7.1 illustrates the difference of deterioration trajectory a vehicle can follow if it has the configuration 1 (top subfigure) or configuration 2 (bottom subfigure).

A maintenance operation brings the vehicle health state back to 0, as it is a perfect maintenance operation.

7.4 The static case

Studying the static joint scheduling problem for missions and maintenance to manage a whole fleet of vehicle is a necessary step to solve the dynamic joint scheduling problem afterwards. The interest in this case is to see the benefit offered by the fleet capacity to schedule and perform all the requested missions. Indeed, a fleet brings flexibility to assign the different missions to the vehicle having the most convenient health state.

This section briefly describes the resolution approach by emphasizing on the differences between the genetic algorithm developed in Chapter 5 to solve the static joint scheduling problem for one vehicle and the one developed to solve the similar problem but with a fleet.

Then, application examples are presented to illustrate the differences of performances when we consider each genetic algorithm to schedule a set of n missions for a fleet of N_v vehicles. Scheduling the fleet activities by using the algorithm designed for a single vehicle suggests to start by splitting the set of missions into as many disjointed subsets as there are vehicles in the fleet. One subset is then assigned to each vehicle and the scheduling algorithm for a single vehicle is applied on each vehicle independently. One of the example considers an homogeneous fleet of vehicles while the other studies the joint scheduling problem for an heterogeneous fleet.

The study presented on the homogeneous fleet is based on [139].

7.4.1 Resolution approach description

Solving the joint scheduling problem for a fleet has some similarities with solving it for one vehicle. However, taking into account the fleet dimension adds a significant complexity level. The question is not just about scheduling the missions in the right order and the maintenance at the right time, but also to assign the missions to the right vehicles.

We firstly introduce the considered optimization criterion and present the differences between the GA for one vehicle and the developed GA for the fleet in the static case.

7.4.1.1 Decision criterion

The schedule optimization is based on the global maintenance costs for the fleet. Missions are then arranged between the different vehicles to minimize the need of maintenance operations and then to minimize the maintenance costs.

The decision criterion C_m^{fleet} is used to manage the decision-making process and obtain the best schedule to minimize the global maintenance costs for the fleet.

The criterion C_m^{fleet} for the schedule Π of all the activities of a fleet composed of N_v vehicles is defined in Eq.7.2. Each vehicle schedule π_v is composed of $N_b(v)$ blocks of missions separated by preventive maintenance operations.

$$\begin{aligned} C_m^{fleet}(\Pi) &= \sum_{v=1}^{N_v} C_1(\pi_v) \\ &= \sum_{v=1}^{N_v} \left(\sum_{k=1}^{N_b(v)} \left(C_0 + C_f \mathbb{P}_f(v, k) \right) \right) \end{aligned} \quad (7.2)$$

There are as many preventive maintenance operations as the number of blocks composing a vehicle schedule and each one of them costs C_0 .

During the missions execution, failures can occur. As we model the deterioration evolution for a whole block rather than mission by mission, we need to estimate the probability of failure occurrences in the block. This probability is denoted $\mathbb{P}_f(k, v)$ in Eq.7.2 and stands for the probability to have a failure in the block k for the vehicle v . It is used to estimate the corrective maintenance costs associated with the potential failure occurrences. The cost of a corrective maintenance is equal to C_f that we assume greater than C_0 due to the immobilization costs associated with a breakdown. Actually, the criterion C_m^{fleet} is an application of the criterion C_1 (Eq. 5.10), introduced in Chapter 5, to the fleet problem.

7.4.1.2 Description of the GA difference between one vehicle and a fleet

To jointly schedule the missions and the maintenance operations for a fleet of vehicles, we propose a genetic algorithm-based methodology. This methodology is not fundamentally different from the one developed in Chapter 5. The differences lie in the characterization of the individual for the genetic algorithm and an adjustment of the different genetic operators to fit to this new individual definition.

Individual representation

The individuals define potential scheduling solutions for missions and maintenance. In Chapter 5, an individual was obtained by distributing the different missions into blocks. In this case, the mission set is split up among the vehicles and, for each vehicle, the mission subset is divided into blocks separated by preventive maintenance operations. Eq.7.3 shows an example of individual Π for a fleet of three vehicles and the set of 20 missions is dispatched between them.

$$\Pi = \begin{cases} \pi_1 = [10, 2] [5] [3, 8, 1] \\ \pi_2 = [12, 20, 16] [4, 6] [15] \\ \pi_3 = [7, 11, 18, 14] [13, 17, 19, 9] \end{cases} \quad (7.3)$$

Initial population

Either when we generate random individuals or when we use heuristic methods (FF, FFD, BFD), we start by generating a single sequence of blocks that we fill with the missions. The second stage enables to split the block sequence between the different vehicles to obtain an individual identical to Π in Eq.7.3.

Crossover and Mutation operators

For the crossover stage, pairs two individuals referred to as parents have been selected in the previous selection stage. We do the crossovers vehicle by vehicle i.e. the schedule for vehicle 1 from parent 1 is crossed with the schedule for vehicle 1 of parent 2. It gives two schedules for the vehicles 1 of the two children. The crossover principle is then similar to the one described in Section 5.2.3.2. At the end, we obtain two new solutions composed of three

sub-schedules.

On the contrary, we concatenate the three sub-schedules of the selected individual before applying the mutation operator. The blocks are marked to remember to which vehicle they belong as well as their position in the vehicle schedule. Then, the mutation operator is applied (cf [Section 5.2.3.2](#)). The last stage consists in recomposing the individual to give it back its original shape.

For the mutation operator, we also define a probability of mutation for each gene p_{mut} i.e. each mission in the schedules.

Stopping criteria

In the literature, the stopping criterion is either defined by a maximum number of iterations, an elapsed time or when there is no change in fitness for a specified number of iterations [156].

In our previous genetic algorithms, we use a maximum number of iterations as stopping criterion. In this case, we adopt the no change in fitness strategy. The GA stops when there is no change of the population's best fitness value for a fixed number of iterations I_{max} . As a very slight change in the best fitness value forces the stopping process to start from scratch, we also add a stopping criterion based on a maximum number of iterations $i_{max} = 1000$. We deliberately choose a high value of i_{max} to favour the use of the first stopping criterion based on the best fitness value.

7.4.2 Performance of the fleet static scheduling approach

To estimate the performance of the fleet static scheduling approach, we define some fleet scheduling application examples to solve with both the fleet static scheduling algorithm and with the single vehicle static scheduling algorithm.

We call in the remaining part of this section:

- **Fleet method:** the static scheduling method based on the fleet genetic algorithm.
- **1VS1 method:** the static scheduling method based on the one vehicle genetic algorithm.

To assess the performance, we compare the two methods to schedule the missions and maintenance operations for a fleet. A process is established ([Figure 7.2](#)); which consists of different steps:

- **1VS1 method:** The first stage consists in randomly assigning the missions to the vehicles so that they may not have the same number of missions to complete. Then, the schedule for each vehicle of the fleet is generated by applying the 1VS1 method to

set of randomly assigned mission. Each random selection and distribution of missions among the vehicles is part of a scenario.

- **Fleet method:** The schedule for each vehicle of the fleet is obtained by applying the fleet genetic algorithm.
- **Monte Carlo simulations:** 1000 Monte Carlo simulations are generated for each scenario to analyse the behaviour of each scheduling method. Naturally, for each simulation, a set of deterioration trajectories is generated. Each trajectory follows the Gamma process $Ga(\alpha_m, \beta_m)$ characterizing each mission m .

This strategy is repeated N_S times to have different random assignments of missions for the 1VS1 method and be able to reach a global maintenance cost convergence for this method.

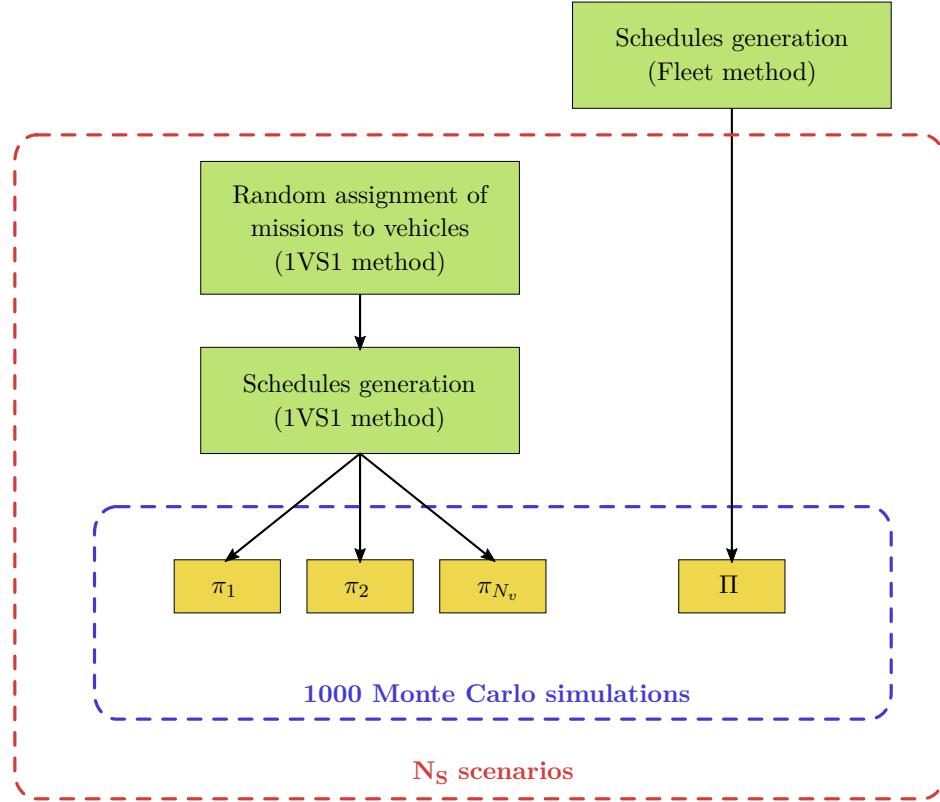


Figure 7.2: Process to compare the fleet and the 1VS1 methods

We present in this section, two application examples. The first one is for an homogeneous fleet of vehicles while the second is for an heterogeneous one.

7.4.2.1 Example for an homogeneous fleet of vehicles

A fleet of $N_v = 2$ vehicles has to complete a set of $n = 12$ missions. All the vehicles are identical and can be dispatched on all the missions. We are in a static case so neither

monitoring information nor any occurring events can change the initial schedules for the vehicles. Dataset E (Table 7.1) describes the missions features.

Table 7.1: Dataset E

Mission	Duration (h)	α_m	β_m	Failure probabilities
1	3	4.17	0.25	0.0021
2	6	4.08	0.35	0.026
3	9	1.62	0.27	0.0034
4	4	5.45	0.33	0.016
5	3	5.61	0.29	0.0056
6	4	4.81	0.31	0.0094
7	3	0.667	0.01	4.99×10^{-4}
8	3	0.0067	0.01	0.0045
9	6	0.213	0.08	6.88×10^{-4}
10	2	0.36	0.06	0.0011
11	2	0.16	0.04	0.0022
12	5	0.036	0.03	0.0032

This example aims at showing that considering the fleet dimension in the scheduling algorithm, rather than scheduling independently the activities for each vehicle, is beneficial to reduce the global maintenance costs for the fleet.

Table 7.2 defines the parameters for the fleet static genetic algorithm.

Table 7.2: Parameters definition for the GA

Parameters	Values	Parameters	Values
C_0	1000	C_f	3000
N_{pop}	200	p_{mut}	0.3
P_{cross}	0.8	P_{mut}	0.5
i_p	4	i_{max}	1000
a	20%	b	20%
ε_{min}	10	ε_{max}	60

• Maintenance costs analysis

The schedules definition is constrained by the maximum failure probability for the blocks of missions \mathbb{P}_{max} . Simulations are then realized for different values of \mathbb{P}_{max} .

The results of the comparison between the fleet and the 1VS1 methods are presented in Table 7.3. It shows that making joint decisions to schedule the missions and maintenance

operations for the whole fleet enables to reduce the maintenance costs. The global maintenance costs gain earned by the fleet method with respect to the 1VS1 method varies between 2% and 6.5% according to the value of \mathbb{P}_{max} . The maximum gain is obtained for $\mathbb{P}_{max} = 0.1$. The last column of Table 7.3 is computed as follows:

- For each scenario, the difference between the average maintenance costs obtained when applying the fleet method and the average maintenance costs obtained when applying the 1VS1 method is computed. It is denoted ΔC_m (Eq.7.4). These average maintenance costs are the mean value on the 1000 Monte Carlo simulations.
- We then count the number of scenarios n_s for which this difference is either greater than 0 or less than 0. If this difference is less than 0, it means that the maintenance costs when using the fleet method are smaller than the ones obtained by using the 1VS1 method. So, the fleet method wins. On the contrary, if the difference is greater than 0, the 1VS1 method wins.
- The last stage is to divide n_s by the total number of scenarios N_S and to convert it in percent.

Table 7.3: Maintenance costs results when comparing the two methods

\mathbb{P}_{max}	Savings with fleet method	Method	C_m	Cases when method wins
0.1	6.45%	Fleet	7254	100%
		1VS1	7722	0%
0.2	2.53%	Fleet	7275	83.4%
		1VS1	7459	16.6%
0.3	2.41%	Fleet	7276	86%
		1VS1	7451	14%

Figure 7.3 represents the distribution of the difference of maintenance costs ΔC_m between the fleet and the 1VS1 methods (Eq.7.4) for different values of \mathbb{P}_{max} . The considered maintenance costs for each method to compute the difference are the costs obtained for the N_S considered scenarios. Remember that those costs are the average maintenance costs on the 1000 Monte Carlo simulations.

The y-axis represents the number of scenarios n_s over the total number of scenarios N_S .

$$\Delta C_m = C_m^{fleet} - C_m^{1VS1} \quad (7.4)$$

When $\mathbb{P}_{max} = 0.1$, the fleet method always wins against the 1VS1 method as ΔC_m is always negative. When \mathbb{P}_{max} increases, we find more scenarios for which ΔC_m is positive. Remember that for the 1VS1 method, we do not apply the scheduling algorithm on the whole set of missions but on different subsets of missions. These subsets are the results of the random assignment stage to define which vehicle is going to execute each mission. The algorithm still

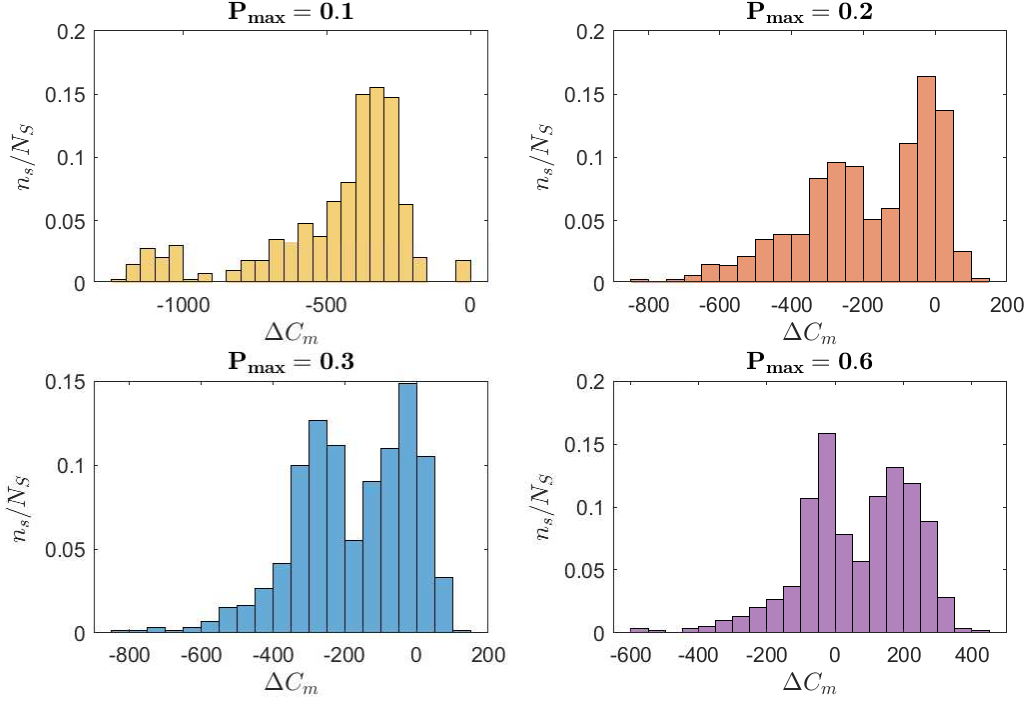


Figure 7.3: Global maintenance cost gain earned when applying the fleet method for different values of P_{\max}

tries to fill the blocks at their maximum while respecting the constraint P_{\max} but may not reach the best blocks definition as it can only group the missions that belong to the same subset. For instance, missions 2 and 8 are assigned to the vehicle A and mission 5 is assigned to the vehicle B . The best grouping strategy would be to define a block with these three missions to have a failure probability for this block really close to P_{\max} . However, it is not possible as mission 5 is not assigned to the same vehicle. The algorithm applied on vehicle A will group at least missions 2 and 8 together and may or may not complete the block with other missions assigned to vehicle A . But this block will have a smaller failure probability than the one for the block composed of missions 2, 5 and 8. So the failure risk is lessened. On the contrary, with the fleet method, the scheduling algorithm will tend to group missions 2, 5 and 8 together and assign the whole block to a vehicle. It then leads to a higher failure risk. This can explain why the fleet method wins only in about 85% of the cases for $P_{\max} = 0.2$ and $P_{\max} = 0.3$.

This general analysis shows the potential maintenance costs that a fleet method can save in average. It can also be interesting to study more precisely the schedule characteristics when the fleet method does not win.

- **Study of specific cases**

The schedules difference between the ones obtained with the fleet method and the ones

obtained with the 1VS1 method comes from the random assignment of the missions to each vehicle for the 1VS1 method and the value of \mathbb{P}_{max} . Indeed, as explained previously, when \mathbb{P}_{max} is higher, the constraint on the way to build the blocks of missions is relaxed so missions grouping becomes easier for the 1VS1 method despite the random mission assignment.

Different observations can be made. When all the missions have similar characteristics (duration, deterioration properties), using the fleet dimension to schedule the fleet activities is not necessary. Then, the gain from one random mission assignment to another can vary a lot (Table 7.4).

When $\mathbb{P}_{max} = 0.2$, many schedules obtained with the 1VS1 method are composed of $N_b = 6$ blocks of missions like the schedule obtained with the fleet method. Some can have smaller maintenance costs than the ones for the fleet method. In these cases, the blocks are less filled with missions that when using the fleet method. So when deterioration trajectories are generated, if a mission, belonging to a very filled block from the fleet schedule, has a huge impact on the vehicle deterioration, it is more likely to have a failure. The fleet method takes the maximum risk to define the schedule to tend towards \mathbb{P}_{max} . The results are summarized in Table 7.4. The values in the last column of the table are positive when the 1VS1 method wins against the fleet method and enables to obtain smaller maintenance costs. For the fleet schedule, the schedule on the left side of $||$ is the schedule for vehicle 1 and the one on the right side is for vehicle 2.

Table 7.4: Comparison of the gains between fleet and 1VS1 schedules examples

\mathbb{P}_{max}	Method	Schedule	N_b	Gain/loss with fleet method
0.1	Fleet	[1, 7][2, 8][4, 12] [6, 11][5, 10][3, 9]	6	
	1VS1	[7, 10, 12] [8, 5][4][11, 3][2][6][1, 9]	7	4.93%
		[4][3][7] [9, 8, 10, 12, 11][5][2][6][1]	8	14.83%
		[5, 10][1, 7] [11, 6][2, 8][9, 3][12, 4]	6	0%
0.2	Fleet	[4][3, 7][2] [11, 6][1, 12, 8, 10][9, 5]	6	
	1VS1	[7, 10][4] [3, 9][6][2][11, 1, 8][5, 12]	7	6%
		[4][1] [9, 7, 10, 11, 8, 12][3][2][6][5]	7	4.28%
		[8, 11, 6][4] [10, 3, 12][7, 1][2][5, 9]	6	-1.29%

This comparison shows the interest of integrating the fleet dimension when scheduling the missions and the maintenance operations for a fleet of vehicles. In most cases, the fleet method enables to reduce the global maintenance costs but on some specific cases, these costs can be slightly higher than with the 1VS1 method because of \mathbb{P}_{max} . The average gain may not seem significant. Considering different vehicle configurations could highlight the interest of using the fleet dimension. We are going to study the static joint scheduling problem for an heterogeneous fleet in the next section.

7.4.2.2 Example for an heterogeneous fleet of vehicles

A fleet of $N_v = 4$ vehicles has to complete $n = 24$ missions. This fleet is composed of two vehicle configuration types: type 1 and type 2. According to the kind of vehicle completing a mission, the mission impact on its deterioration is different. Dataset F, presented in Table 7.5, defines the missions for each type of vehicle. All the vehicles can perform all the missions no matter the impact of the mission on their deterioration. Note that there are as many missions having a high impact on the type-1 vehicles deterioration as missions having a high impact on the type-2 vehicles deterioration. For instance, mission 2 in dataset F has a higher impact on vehicle of type 2 whereas it has a low impact on vehicle of type 1. It is the contrary for mission 4.

The objective of this example is to show that when we have an heterogeneous fleet, jointly scheduling the activities for all the vehicles at the same time has an advantage. Indeed, it enables to consider the vehicle configuration and the effect of the missions on its deterioration. The problem to solve is then about choosing the right vehicle for the right usage conditions to minimize its deterioration evolution and avoid additional maintenance operations.

Table 7.5: Dataset F

Mission	Duration	VI type 1			VI type 2		
		α	β	\mathbb{P}	α	β	\mathbb{P}
1	3	4.17	0.25	0.0021	0.67	0.10	0.0005
2	6	0.01	0.01	0.0091	4.08	0.35	0.0261
3	9	0.21	0.08	0.0026	1.62	0.27	0.0034
4	4	5.45	0.33	0.0157	0.36	0.06	0.0066
5	3	5.61	0.29	0.0056	0.16	0.04	0.0044
6	4	0.04	0.03	0.0024	4.81	0.31	0.0094
7	12	1.56	0.34	0.0018	0.04	0.06	0.0009
8	10	0.10	0.08	0.0004	0.30	0.08	0.0092
9	9	1.06	0.20	0.0043	3.35	0.49	0.0015
10	7	1.61	0.23	0.0019	4.20	0.46	0.0034
11	9	2.91	0.47	0.0003	1.89	0.32	0.0013
12	15	1.45	0.39	0.0011	0.17	0.13	0.0001
13	5	1.42	0.18	0.0008	3.47	0.26	0.0260
14	6	2.48	0.27	0.0043	2.40	0.30	0.0005
15	6	3.87	0.33	0.0343	1.17	0.14	0.0106
16	13	1.08	0.28	0.0016	0.64	0.16	0.0142
17	14	1.03	0.25	0.0096	2.38	0.46	0.0247
18	7	2.77	0.33	0.0047	2.67	0.34	0.0020
19	10	0.10	0.09	0.0001	3.19	0.50	0.0030
20	9	0.48	0.13	0.0012	1.58	0.32	0.0002
21	11	0.51	0.13	0.0055	2.88	0.46	0.0097
22	14	1.19	0.32	0.0013	2.33	0.48	0.0089
23	11	1.64	0.26	0.0466	1.69	0.34	0.0013
24	11	0.27	0.10	0.0034	1.96	0.40	0.0005

- Maintenance costs analysis

The results presented in Table 7.6 shows that the maintenance costs savings generated when using the fleet method increase when we consider an heterogeneous fleet of vehicles. It enables to generate around 25% maintenance costs savings. A scheduling algorithm that coordinates the missions dispatching and maintenance decisions at a fleet level rather than for each vehicle independently is then very beneficial to reduce the global fleet maintenance costs.

We can observe that the maintenance costs savings slightly increase for the 1VS1 methods when 75% of the vehicles are of type 2. Even though there is the same number of missions having a high impact on each vehicle type deterioration. However, a change in the composition of the fleet does not significantly reduce the maintenance costs gain. It probably comes from the fact that the impact on the deterioration for a mission dispatched on a type-1 or type-2 vehicle is not high enough to see such a change.

Table 7.6: Maintenance costs convergence when comparing the fleet and the 1VS1 methods for different values of \mathbb{P}_{max} and different fleet composition

\mathbb{P}_{max}	Distribution VI type	Savings fleet method	Method	C_m	Method wins
0.3	50% type 1 - 50% type 2	24%	Fleet	14577	100%
			1VS1	19190	0%
	75% type 1 - 25% type 2	25.1%	Fleet	14654	100%
			1VS1	19560	0%
	25% type 1 - 75% type 2	26.6%	Fleet	14893	100%
			1VS1	20299	0%
0.2	50% type 1 - 50% type 2	26.8%	Fleet	14572	100%
			1VS1	19901	0%
	75% type 1 - 25% type 2	25.3%	Fleet	14645	100%
			1VS1	19605	0%
	25% type 1 - 75% type 2	26.9%	Fleet	14904	100%
			1VS1	20379	0%
0.1	50% type 1 - 50% type 2	27.1%	Fleet	14948	100%
			1VS1	20497	0%
	75% type 1 - 25% type 2	24.5%	Fleet	15311	100%
			1VS1	20290	0%
	25% type 1 - 75% type 2	25.7%	Fleet	15464	100%
			1VS1	20819	0%
0.05	50% type 1 - 50% type 2	21.9%	Fleet	16625	100%
			1VS1	21286	0%
	75% type 1 - 25% type 2	21.9%	Fleet	16627	100%
			1VS1	21285	0%
	25% type 1 - 75% type 2	22.5%	Fleet	16637	100%
			1VS1	21479	0%

Figures 7.4, 7.5 and 7.6 show the distributions of the maintenance costs difference between the two scheduling methods ΔC_m . To simplify the description of the histograms, we denote the fleet composition as follows.

- **Fleet 50-50:** the fleet composed of 50% of its vehicles for each range.
- **Fleet 75-25:** the fleet composed of 75% of type-1 vehicles and 25% of type 2.
- **Fleet 25-75:** the fleet composed of 25% of type-1 vehicles and 75% of type 2.

When $\mathbb{P}_{max} = 0.05$, some peaks are visible around the multiples of the preventive maintenance costs $C_0 = 1000$ and corrective maintenance cost $C_f = 3000$. It means that the fleet method generates schedules either with less blocks or that enables to avoid failures. In addition, for \mathbb{P}_{max} equal to 0.3, 0.2 and 0.1, the histograms for the fleet 50-50 seem normally distributed around $\Delta C_m = -5500$ while the ones for the fleet 75-25 are slightly shifted at the right of $\Delta C_m = -5000$ and the ones for the fleet 75-25 shift towards the right of $\Delta C_m = -5500$. The ranges of ΔC_m values where the histograms are the most dense are:

- **For the fleet 50-50:** For all the values of \mathbb{P}_{max} , the major part of the histograms are in the interval $[-6000; -4000]$.
- **For the fleet 75-25:** The values of ΔC_m are mostly between -6000 and -3500 .
- **For the fleet 25-75:** The range of values for ΔC_m is more in $[-7000; -4500]$

It then seems more relevant to use a fleet composed of 75% of vehicles with type 2 to complete the missions set defined in Table 7.5 to reduce the maintenance expenses.

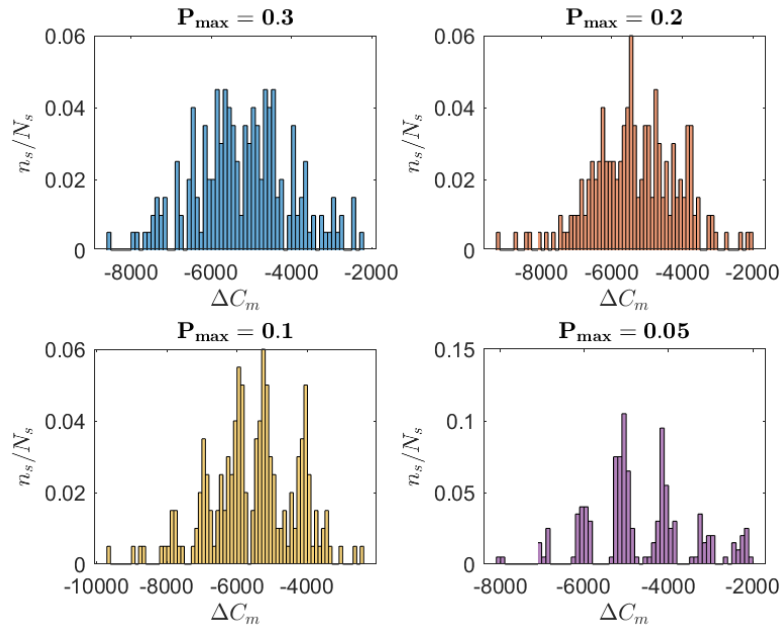


Figure 7.4: Distribution of ΔC_m for a fleet with 50% of the vehicles for each vehicle type

The presented examples manage to show that making decisions at the fleet level can significantly help reducing the expenses spent to maintain the vehicles of the fleet and complete

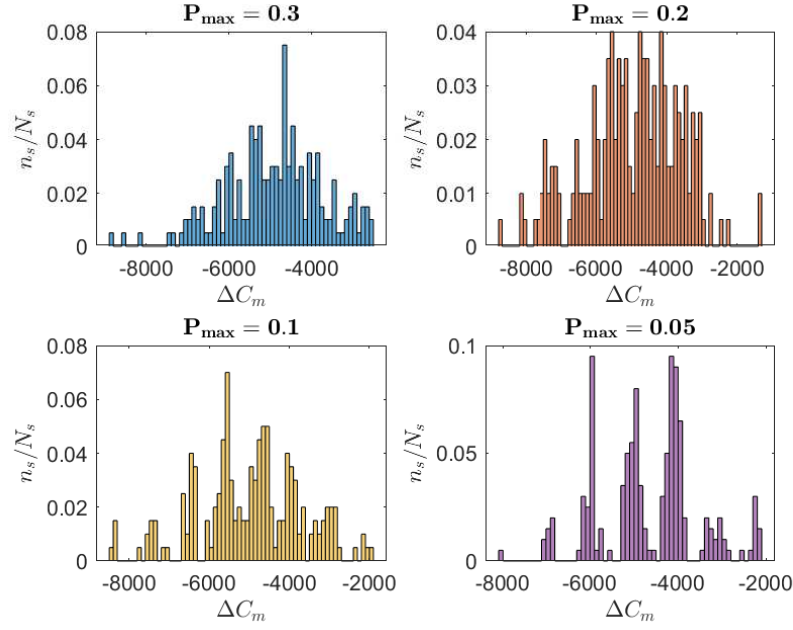


Figure 7.5: Distribution of ΔC_m for a fleet with 75% of the vehicles with type 1 and 25% with type 2

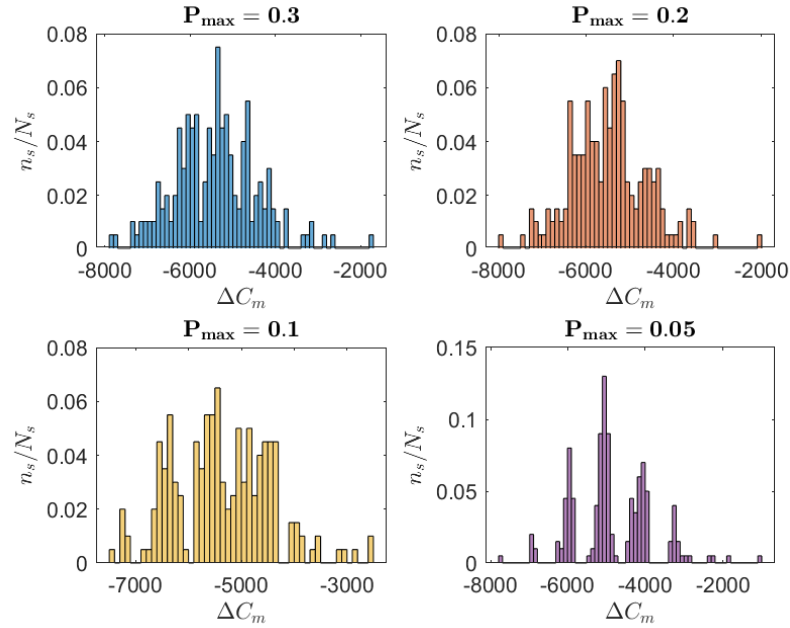


Figure 7.6: Distribution of ΔC_m for a fleet with 25% of the vehicles with type 1 and 75% with type 2

the required missions. It enables to make the decisions for one vehicle according to the state of the rest of the fleet. This observation could be emphasized in a dynamic framework when

information about the fleet state are available to update the initial schedule accordingly. This is the study purpose of the following section.

7.5 The dynamic case

This section proposes a methodology to solve the joint scheduling problem of missions and maintenance for a fleet of vehicle. This method adapts the decision-making process to the available real time information regarding the different vehicles to update the schedules for every vehicle. They are three events to consider for potential schedule updates: new requested missions, failure occurrences and deterioration measurements.

The objective is to mix both the fleet and the dynamic aspects to make the best possible decisions to dispatch the missions on the right vehicle and schedule the maintenance operations according to the health state evolution of the vehicle.

In a first step, the resolution approach is described to understand how the missions and maintenance schedule for all the vehicles are defined and updated based on the available monitoring information. The approach principle will be detailed on a example. Then, an application example is presented to compare the method developed in [Chapter 6](#) that schedule the activities for the trucks independently and the fleet dynamic method developed in this section.

7.5.1 Resolution approach description

The resolution approach is based on the one described in [Section 6.2](#) for the dynamic method for a single vehicle.

This section begins with a brief reminder of the definition of the missions before digging into the designed scheduling decision-making process. To design a decision-making process, the first stage is to define an optimization criterion. Then, the fleet dynamic scheduling methodology is explained.

7.5.1.1 Definition of the missions features

The missions the vehicles have to complete are characterized by specific operating conditions that represent their usage severity. It defines the impact they will have on the vehicles deterioration evolution. As the vehicles have different configuration, the usage severity will be different from one vehicle configuration to another.

A mission m is then characterized by:

- $d(m)$: the duration necessary to complete it;

- $(\alpha_{m_y}, \beta_{m_y})$: the shape and scale parameters for the Gamma process modelling the deterioration of a vehicle with the configuration y . If there are two configurations of vehicles, the mission will then be associated to two pairs of parameters, one for each configuration;
- g_m : the raw gain earned when the mission m is completed;
- $c_d(m)$: the unitary delay cost;
- $d_{max}(m)$: the deadline of the mission i.e. the date before which the mission has to be finished. This feature is essential because it represents the priority of the mission. During a rescheduling, a mission can be postponed to prioritize a mission with a closer deadline to avoid delay penalty costs.

Note that in this case, we consider the deadline and not the starting deadline as in [Chapter 6](#).

We assume in the rest of this section that all the missions have the same raw gain g_m and the same unitary penalty cost c_d .

7.5.1.2 Decision criterion

The decision-making process to schedule the missions and the maintenance operations for the fleet of vehicles aims at optimizing the operating incomes of the fleet. The criterion I_o (Eq.7.5) is then defined to estimate the operating incomes for the fleet schedule Π .

$$I_o(\Pi) = G_p - C_m - C_d - C_l \quad (7.5)$$

The operating incomes corresponds to the difference between the gain generated by completing the missions G_p and the sum of three different costs C_m , C_d et C_l . They represent:

- C_m : maintenance costs for the fleet;
- C_d : delay penalty costs if the missions are finished after their deadlines;
- C_l : loading/unloading penalty costs that is induced when an on going mission on a vehicle is assigned to another vehicle after a rescheduling.

G_p corresponds to the sum of all the raw gains g_m of the missions to complete. It is then a constant value.

- **Maintenance costs C_m**

The maintenance costs for the fleet is the sum of the maintenance costs for each vehicle. It includes the preventive maintenance costs induced after the preventive operations at the end of every block of missions and the corrective maintenance costs.

To estimate the corrective maintenance costs, we consider one failure by block of missions.

The global maintenance costs C_m is then defined by Eq.7.6.

$$C_m = \sum_{v=1}^{N_v} \left(\sum_{b=1}^{N_b(v)} \left(C_0 + C_f \cdot \mathbb{P}_f(b, v) \right) \right) \quad (7.6)$$

where C_0 and C_f are respectively the preventive and corrective costs for a maintenance operation, N_v is the number of vehicles in the fleet, $N_b(v)$ is the number of blocks in the schedule of vehicle v and $\mathbb{P}_f(b, v)$ is the probability to have one failure in the block b of the vehicle v schedule.

- **Delay penalty costs C_d**

The penalty costs associated with delay in the missions C_d is computed by estimating the potential delay time of each mission and multiplying it by the unitary delay cost of each mission (Eq.7.7).

$$C_d = \sum_{j=1}^n t_d(j) \times c_d(j). \quad (7.7)$$

The delay time $t_d(i)$ for the mission i is the difference between the time $t(i)$ when the mission i ends and the deadline $d_{max}(i)$.

We assume that mission i belongs to block k in the schedule of vehicle v . The estimation of $t(i)$ is done as follows.

- * Estimation of the time spent in maintenance before the beginning of the block containing the mission i denoted $T_M(i)$ (Eq.7.8).

We need to know the number of preventive maintenance operations performed before block k and the number of failure occurrences. As mission i is in block k , $k-1$ preventive maintenance occurred. Each operation lasts d_p . The estimated number of failure occurrences in a block b is denoted $\mathbb{E}[N_f(b)]$ and each corrective operation lasts d_c .

$$T_M(i) = (k-1)d_p + \sum_{b=1}^{k-1} d_c \mathbb{E}[N_f(b)] \quad (7.8)$$

- * Estimation of the time spent in mission until mission i , including the duration of mission i , $T_{in_m}(i)$.

This time corresponds to the sum of all the durations of the missions j in the blocks before block k and the durations of the ones until mission i included in block k .

$$T_{in_m}(i) = \sum_{b=1}^{k-1} \sum_{j \in B_b} d(j) + \sum_{j \in B_k | j \leq i} d(j) \quad (7.9)$$

- * Estimation of the time spent in corrective maintenance from the first mission belonging to the block containing mission i until mission i , mission i included, denoted $T_{CM}(i)$.

An estimation of the number of failures occurrences during these missions is required. $\mathbb{E}[N_f(k|j \leq i)]$ denotes the estimation of this number of failure occurrences.

$$T_{CM}(i) = d_c \mathbb{E}[N_f(k|j \leq i)] \quad (7.10)$$

The time $t(i)$ when mission i ends is then estimated as in Eq.7.9.

$$t(i) = t_0 + T_M(i) + T_{in_m}(i) + T_{CM}(i) \quad (7.11)$$

where t_0 is the initial time when the current schedule has been generated.

The delay time $t_d(i)$ is then equal to 0 if the time when the mission i ends $t(i)$ is smaller than its deadline $d_{max}(i)$. Otherwise, the delay time is the difference between $t(i)$ and $d_{max}(i)$.

- **Loading/unloading penalty costs C_l**

A loading penalty cost c_l is charged when, after a rescheduling, a mission that started being executed on a vehicle A is now assigned to another vehicle of the fleet. After a rescheduling, the maximum value of the loading/unloading costs is $c_l \times N_v$, with N_v the number of vehicles in the fleet. If the on going mission has just been postponed to prioritize another mission but is still assigned to the same vehicle, this cost c_l is not charged.

7.5.1.3 Description of the rescheduling method for the fleet

The rescheduling method for the fleet is based on a genetic algorithm designed as a combination of the genetic algorithm applied for the dynamic joint scheduling problem for one vehicle and the genetic algorithm used to solve the static joint scheduling problem for a fleet.

At the beginning, an initial schedule Π is generated using this genetic algorithm. This schedule defines the activities to do for the different vehicles composing the fleet. Once Π is implemented and its execution starts, sequential rescheduling can occur due to three types of real time events: failure occurrences, deterioration measurements and new requested missions. The principle is described in .

- **Failure occurrence:** A failure occurrence during a mission implies to perform a corrective maintenance operation to restore the vehicle to its initial health state, equal to 0. A cost C_f is then charged for the maintenance. When a vehicle has a failure, a rescheduling is immediately triggered and executed on the whole fleet and not just on the vehicle affected by the failure.
- **New mission:** When a new mission is requested, a rescheduling for the whole fleet is executed. We assume that if the other vehicles are executing a mission, they stop and wait for their new mission order. It enables to integrate the mission in the schedule quickly and avoid at best any potential delay on the mission.
- **Deterioration measurement:** A deterioration measurement on a vehicle gives us an indication on its health state. But a rescheduling is not necessarily triggered. The first element to check is if the vehicle can go on and finish its assigned missions. If it cannot, a rescheduling is triggered to avoid any further failure occurrence. If it can finish the missions, the potential new schedule is still computed. If the difference of operating incomes between the new and the current schedules is greater than ΔC_{min} , the new schedule is adopted and replaces the old ones. Hence, the vehicles have new missions assignments to complete. ΔC_{min} is the minimum income that a rescheduling has to generate to be profitable. Indeed, every time a rescheduling is applied, the whole fleet activity changes and could lead to some serious logistic complications.

Any rescheduling occurrence affects the whole fleet, no matter the activity they were performing at the time. We assume that the vehicles can stop their current missions at any time to take new ones if the rescheduling algorithm defines it as the best decision to optimize the fleet operating incomes. For new missions, it is understandable to reschedule the whole fleet schedule because we do not know yet to which vehicle the mission will be assigned. However, failure occurrences and deterioration information are associated to a specific vehicle at a specific moment. It could then seem more relevant to only reschedule the activity of this specific vehicle.

This is where we can really see the interest of making decision at the fleet level. Indeed, if we only reschedule for the specific vehicle after a failure, a risk of delay appears. By rescheduling the whole fleet schedule, this delay could be avoided if another vehicle finishes the mission assigned to the failed one, as long as it does not jeopardize the completion of its own current mission before its deadline.

If a deterioration information is available and the vehicle has a high risk of failure, a rescheduling will probably send it to maintenance before restarting its missions. On the

contrary, if we reschedule the whole fleet activity, we may find another mission, normally assigned to another vehicle, that it could complete before being maintained to make the most of its remaining useful lifetime.

With the dynamic scheduling method for one vehicle, we had four versions of the genetic algorithm. One for each type of event (failure, deterioration information and new mission) and one for the combination of both deterioration measure and new mission. However, with the fleet dynamic scheduling method, we have to consider all the events simultaneously as they can occur at the same time on different vehicles. The genetic algorithm is then designed to consider all the potential events simultaneously.

It also brings the question of the vehicles availability into the light. Different cases can be listed:

- If any type of event occurs and triggers a rescheduling, all the vehicles dispatched on a mission stop their activity and wait for their new orders. The remaining parts of the on going missions may be assigned to the same vehicle as the first mission to perform after the reschedule. But they could be postponed or event assigned to other vehicles. Vehicles on mission are considered as available as soon a rescheduling is applied.
- If a rescheduling is triggered when some vehicles are performing a maintenance, they are considered as unavailable. They will only be available when the maintenance operation will be completed. The algorithm considers this unavailability constraint to build the new optimized schedule.

Sequential rescheduling of the whole fleet enables to adapt the schedule, the missions order to complete and the maintenance activity according to the health state of all the vehicles.

7.5.2 Application example to analyse the performances of the fleet dynamic scheduling method

An application example is defined to evaluate the performance of the dynamic scheduling method for the fleet and compare it with the use of the single vehicle dynamic scheduling method to schedule the vehicles activities independently.

We consider an heterogeneous fleet of $N_v = 3$ vehicles. Two of them have the configuration 2 while the third one has the configuration one. We denote them V_1 , V_2 and V_3 .

At the beginning, $n = 12$ missions are available (grey parts in [Table 7.8](#)) in the mission pool and have to be completed. Four missions will then be added during the schedule completion. These missions are added to the mission pool only when some other specific missions are over. The missions 13 and 14 are requested once the mission 6 is completed. The missions 15 and 16 are respectively requested when the missions 7 and 3 are finished.

Monitoring information is available at the end of the missions 3,4,8,9 and 13.

Two methods are considered to schedule the fleet activities:

- **1VS1 dynamic scheduling method:** This method consists in, firstly, assigning the missions to the vehicles composing the fleet. Then, for each vehicle independently the dynamic scheduling method for a single vehicle is applied. There is no communication between the vehicles during their schedules completion. It is then a decentralized scheduling strategy. For the missions that are requested later, they are assigned to the vehicles that have to complete the missions that trigger them. For instance, if mission 6 is assigned to the vehicle 3, the missions 13 and 14 will also be assigned to the vehicle 3.
- **Fleet dynamic scheduling method:** This method schedules the maintenance operations and the missions for the whole fleet simultaneously. It takes into account all the available information regarding the vehicles health state to make decisions. It is then a centralized approach.

Table 7.7 presents the parameters values for the fleet dynamic scheduling algorithm.

Table 7.7: Parameters definition for the dynamic fleet GA

Parameters	Values	Parameters	Values
C_0	1000	C_f	3000
C_l	300	I_{max}	60
N_{pop}	200	p_{mut}	0.3
P_{cross}	0.8	P_{mut}	0.5
i_p	4	i_{max}	100
a	20%	b	20%
ε_{min}	10	ε_{max}	60

Table 7.8 defines the features of the missions the fleet has to complete. The pair of parameters (α_m, β_m) are the shape and scale parameters of the Gamma process modelling the deterioration evolution and the probabilities \mathbb{P} are obtained by using the failure deterioration threshold $L = 100$ and the parameters (α_m, β_m) and applying the equation 5.8. $VIttype$ is the configuration of the vehicle.

For this example, the decision variables \mathbb{P}_{max} and ΔC_{min} are respectively fixed at 0.15 and $\frac{C_0}{2} = 500$.

The objective is to study the behaviour of the fleet dynamic scheduling method and compare the obtained global operating incomes for the fleet to see if a centralized scheduling approach is more relevant to increase the operating incomes than a decentralized approach.

We use Monte Carlo simulations to generate different deterioration trajectories associated with the Gamma processes parameters for each mission. It enables to analyse the convergence of the operating incomes for both methods.

Table 7.8: Dataset H

Mission	Duration	g_m	d_{\max}	c_d	VI type 1			VI type 2		
					α	β	\mathbb{P}	α	β	\mathbb{P}
1	3	5000	8	50	4.17	0.3	0.0021	0.67	0.1	0.0005
2	6	5000	25	50	0.01	0.0	0.0091	4.08	0.4	0.0261
3	9	5000	40	50	0.21	0.1	0.0026	1.62	0.3	0.0034
4	4	5000	10	50	5.45	0.3	0.0157	0.36	0.1	0.0066
5	3	5000	12	50	5.61	0.3	0.0056	0.16	0.0	0.0044
6	4	5000	5	50	0.04	0.0	0.0024	4.81	0.3	0.0094
7	12	5000	33	50	1.56	0.3	0.0018	0.04	0.1	0.0009
8	10	5000	50	50	0.10	0.1	0.0004	0.30	0.1	0.0092
9	9	5000	22	50	1.06	0.2	0.0043	3.35	0.5	0.0015
10	7	5000	40	50	1.61	0.2	0.0019	4.20	0.5	0.0034
11	9	5000	35	50	2.91	0.5	0.0003	1.89	0.3	0.0013
12	15	5000	50	50	1.45	0.4	0.0011	0.17	0.1	0.0001
13	5	5000	30	50	1.42	0.2	0.0008	3.47	0.3	0.0260
14	6	5000	30	50	2.48	0.3	0.0043	2.40	0.3	0.0005
15	6	5000	70	50	3.87	0.3	0.0343	1.17	0.1	0.0106
16	13	5000	100	50	1.08	0.3	0.0016	0.64	0.2	0.0142

7.5.2.1 Analysis of the fleet schedule evolution on a simulated scenario

A set of deterioration trajectories has been generated to realize a scheduling simulated case based on the scenario defined previously regarding the new requested missions and the monitoring information. The objective is to schedule and complete the 16 missions and perform the maintenance operations at the right time to avoid failures while taking into account the monitoring information available.

- **Evolution of the schedules for each vehicle**

At the beginning, the algorithm dispatches the 12 available missions among the 3 vehicles. The initial schedule Π_{d_i} is defined in Eq.7.12.

$$\Pi_{d_i} = \begin{cases} \pi_1 = [4, 5, 7, 12] \\ \pi_2 = [6, 2, 3, 8] [10] \\ \pi_3 = [111] [9] \end{cases} \quad (7.12)$$

Table 7.9 and Figure 7.7 describe the schedules evolutions for the three vehicles according to the occurring events. For instance, the first rescheduling is applied when V_2 finishes mission 6 because two new missions, 13 and 14, are requested. It is then a mandatory rescheduling. As a deterioration measurement is available for V_{I_1} , because mission 4 is also completed, that information is integrated as input in the rescheduling algorithm.

Based on Table 7.9 and Figure 7.7, a few comments can be made. Mission 9 switches from VI_3 to VI_2 after a rescheduling. At this rescheduling date $t = 19$, we know that a delay cost for one time unit delay on mission 9 will be charged. We then have to make the best decision based on the fleet state and the other missions situations:

- Missions 13 and 14 will soon reach their deadlines that is equal to 30.
- Based on the obtained new schedule, it seems that VI_3 is in need of a maintenance operation soon.
- VI_2 health state is known as it just completed mission 3.
- Mission 13 has a more severe impact on the deterioration of the vehicles with configuration 2 i.e. VI_1 or VI_3 .
- Mission 14 has a more severe impact on the deterioration of the vehicles with configuration 1 i.e. VI_2 .
- Mission 9 has a more severe impact on the deterioration of the vehicles with configuration 1 i.e. VI_2 .

At that time, it was probably the best decision to avoid more delays on the other missions and avoid any additional maintenance operation in the future with respect to the ones scheduled in this resulting schedule.

The second comment is based on the schedule for VI_1 for which there is an idle time before completing mission 15. Before the rescheduling at $t = 38$, mission 15 was supposed to be dispatched on VI_3 just after mission 12.

Table 7.9: Schedule evolution after each rescheduling

Rescheduling	VI	Schedule	Comments
1: New mission	VI_1	[5, 7][14, 12]	Deterioration measurement at the end of mission 4 before the rescheduling
	VI_2	[2, 3][13, 8]	
	VI_3	[11][9][10]	
2: New mission	VI_1	[][10]	Mission 7 finished before the rescheduling
	VI_2	[9][13, 8][16]	Mission 3 finished before the rescheduling
	VI_3	[14][12, 15]	On going mission 9 not assigned to VI_3
3: Deterioration information	VI_1	[10]	
	VI_2	[13, 8][16]	Deterioration measurement at the end of mission 9 before the rescheduling
	VI_3	[14][12, 15]	
3: Failure	VI_1	[15]	Idle time before the assignment of mission 15
	VI_2	[16]	Failure on VI_2 at the end of mission 8
	VI_3	[12]	

In fact, the probability to have a failure if mission 15 remains in VI_3 is equal to 0.0848 while same probability if the mission is assigned to VI_1 is 0.0106. Both are less than \mathbb{P}_{max} so

the two solutions are feasible. But, in the final schedule, an additional preventive maintenance operation is performed at the end of mission 15 while it would not have occurred if the mission had remained assigned to VI_3 . Actually, for the genetic algorithm, an empty block still counts as a block so a preventive maintenance operation at the end of the block. A schedule keeping mission 15 in the same block as mission 12 means that the schedule for VI_1 is composed of one empty block. The fitness function still charges the preventive cost.

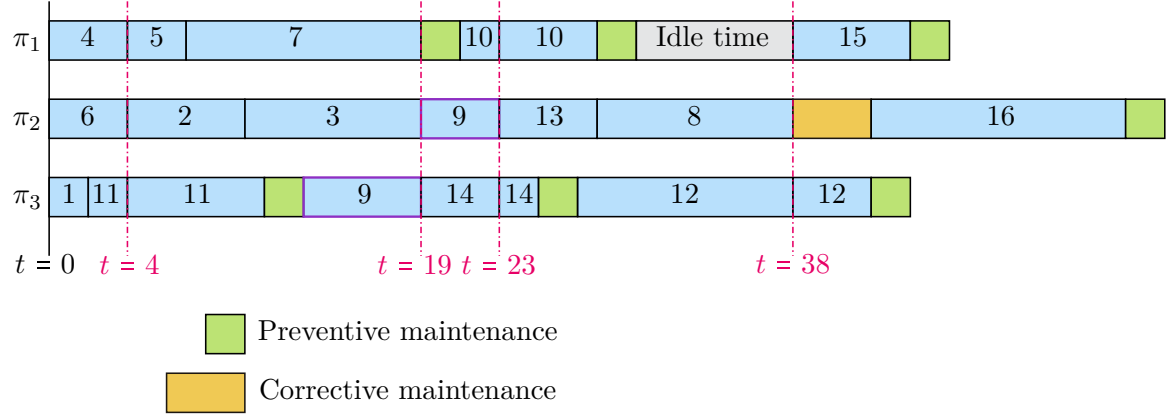


Figure 7.7: Final schedule II followed by the fleet when considering the different schedule updates

Figures 7.8, 7.9, 7.10 illustrate the deterioration evolution for the different vehicles of the fleet according to the events that occurred during the schedule execution.

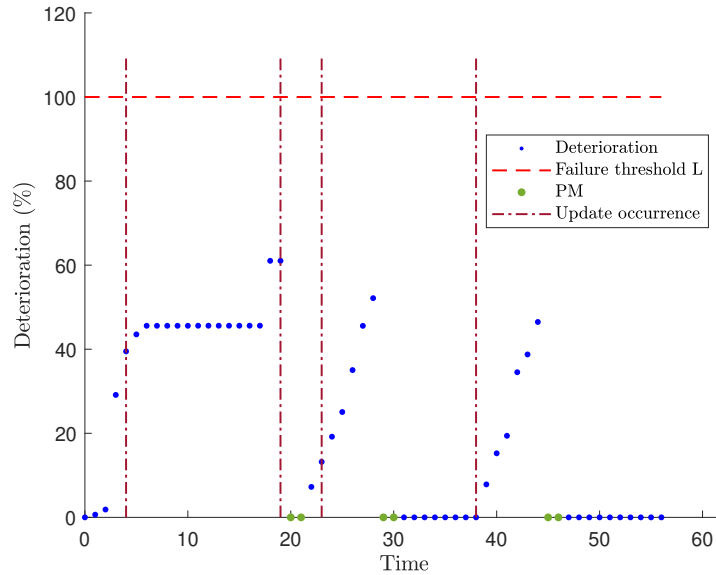
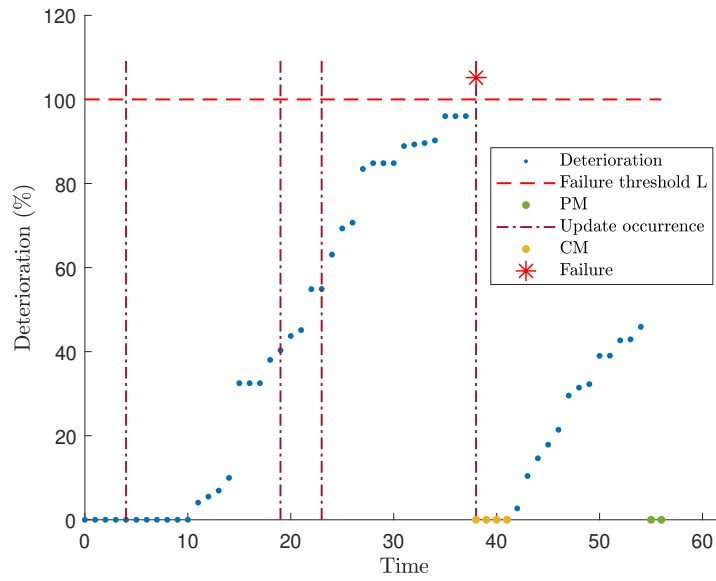
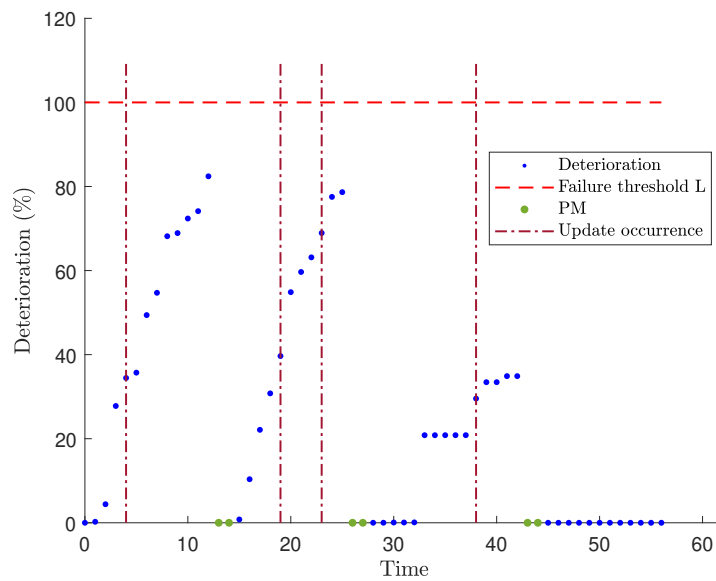


Figure 7.8: Deterioration evolution for VI_1

Figure 7.9: Deterioration evolution for VI_2 Figure 7.10: Deterioration evolution for VI_3

The global operating incomes for the fleet I_o earned with the schedule Π as well as the expenses related to maintenance C_m , delay C_d and switch of on going missions between vehicles C_l are summarized in Table 7.10. $C_d = 50$ because there is only a delay of one time unit for mission 9 with respect to its deadline. $C_l = 300$ is due to the switch of assignment for mission 9 from V_3 to VI_2 .

Table 7.10: Operating incomes and costs distribution

I_o	C_m	C_d	C_l
69650	10000	50	300

7.5.2.2 Performance analysis

To assess the performance of the fleet method, we draw a comparison between the fleet method and the 1VS1 method. We propose two studies.

- **Study 1**

This study is based on the generation of 250 Monte Carlo simulations to define deterioration scenarios. One deterioration trajectory is associated to each mission for each type of vehicle.

For the fleet dynamic method, we use each scenario to schedule the missions and maintenance operations for the fleet. The schedule evolves during its completion to integrate the real time events.

For the 1VS1 dynamic method, we consider a specific assignment of missions to the vehicles. Once a mission has been assigned to a vehicle it cannot switch from one to another. Then, the schedule for each vehicle evolves and is update when real time information requires it. However, a real time information affects only the schedule of the concerned vehicle as the schedules are completely independent. The first stage is to randomly assign the missions to the vehicle. Table 7.11 describes how the missions are dispatched between the vehicles. A first observation is that they are mostly dispatched between VI_2 and VI_3 . VI_1 only has two missions to complete.

Table 7.11: Mission assignment to vehicle

Mission	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
VI	3	2	3	2	3	3	2	1	1	3	3	3	3	3	2	3

Figure 7.11 illustrates the convergence of the operating incomes for the fleet and the 1VS1 dynamic scheduling methods. Note that we remove the costs induced by the switch between vehicles for the fleet method to really compare the same operating incomes based on the gains earned by completing the missions, the maintenance costs and the delay costs. The fleet scheduling method enables to generate schedules that increase the global operating incomes for the fleet of about 12%.

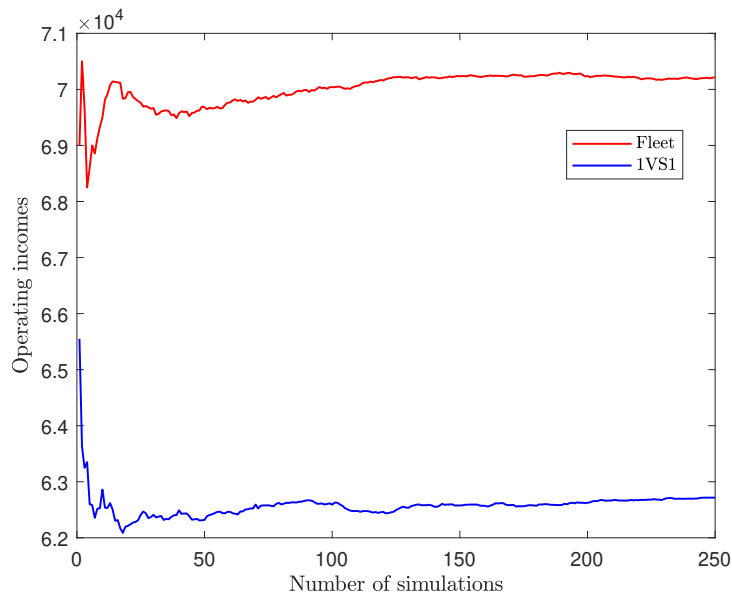


Figure 7.11: Operating incomes convergence for the fleet and the 1VS1 methods

Figure 7.12 shows the distribution of the operating incomes difference between the fleet and the 1VS1 methods. The main part of the simulations show that the schedule generated with the fleet dynamic scheduling method enable to increase the operating incomes from 7000 to 8000. It is the equivalent of the cost of 7 to 8 preventive maintenance operations.

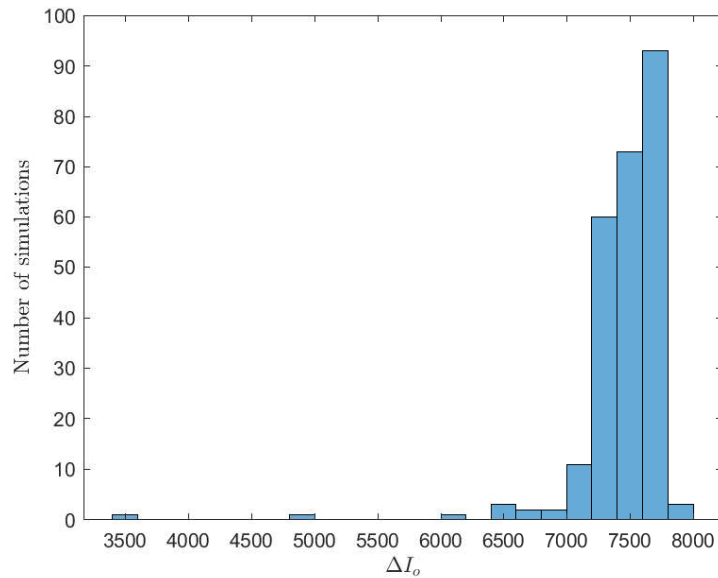


Figure 7.12: Distribution of the operating incomes difference ΔI_o between the two methods for the simulations

From the 250 considered simulations, the schedules obtained with the fleet method always

enable to increase the operating incomes, even if sometimes, they have higher maintenance costs (left subfigure from Figure 7.13). It means that reducing the number of preventive maintenance operations does not help to avoid delays as the delay costs difference is always negative (right subfigure from Figure 7.13).

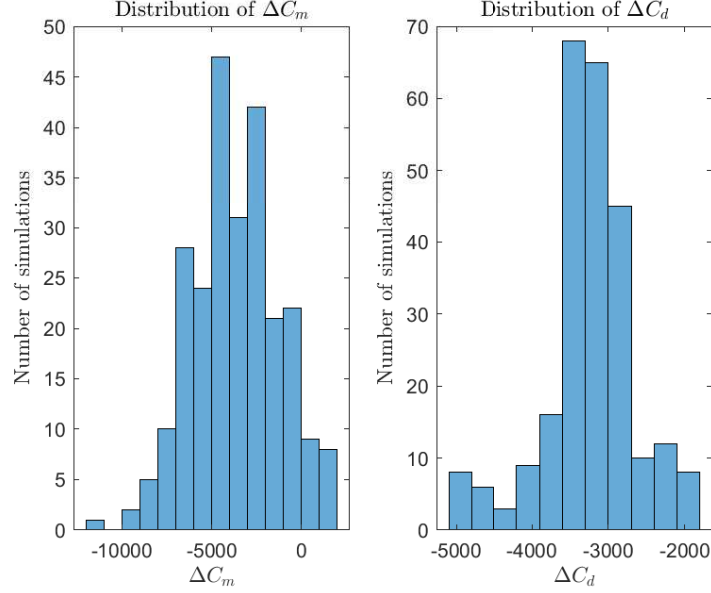


Figure 7.13: Distribution of the maintenance costs and delay costs difference denoted ΔC_m and ΔC_d between the two methods for the simulations

In average, we reschedule as often with the fleet scheduling method than with the 1VS1 scheduling method. For each method respectively, the average values of the total number of rescheduling n_r are 4.6 and 4.4. However, the number of rescheduling due to failure n_f is higher with the 1VS1 scheduling method. There are then more failure occurrences in the schedule obtained with this method. The deterioration information helps more often the fleet method to reschedule as n_d , the number of rescheduling due to deterioration measurements is slightly higher than for the 1VS1 method. Note that sometimes n_{nm} , the number of rescheduling due to new missions, is equal to 2 for the fleet methods. It means that two missions after which new missions are requested are ending at the same time.

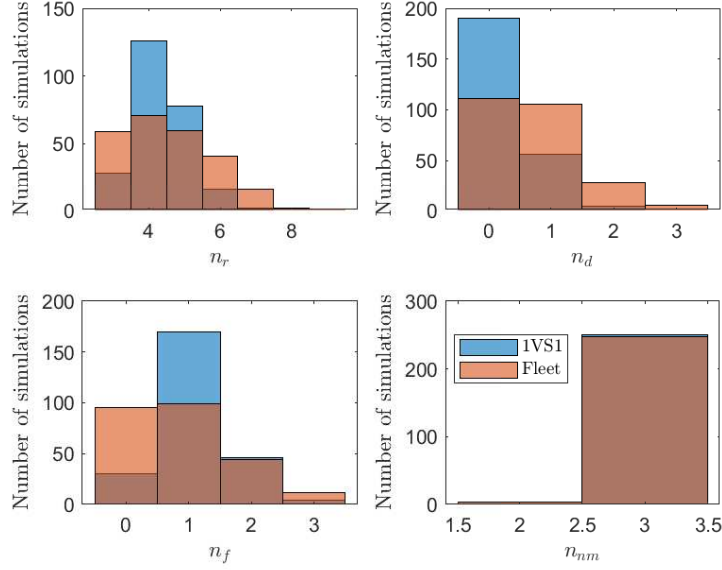


Figure 7.14: Distributions of the total number of rescheduling n_r and the numbers of rescheduling due to deterioration information, failure and new missions respectively denoted n_d , n_f and n_{nm}

• Study 2

This study aims at analysis the operating incomes difference between the fleet and the 1VS1 dynamic method.

For the 1VS1 dynamic method, we define 250 different assignment orders for the fleet. For each assignment, the mission pool is separated into 3 disjointed subsets and each subset is assigned to one vehicle. We then consider only one deterioration scenario for each assignment order.

For the fleet dynamic method, we consider 250 deterioration scenarios to schedule the missions and the maintenance operations.

Based on the 250 scenarios of simulations, the schedules generated by the fleet method enable in average to increase the global operating incomes for the fleet of 6.3% with respect to the ones obtained with the 1VS1 method (Figure 7.15).

Figure 7.16 shows that the operating incomes difference ΔI_o between the two methods show is mostly between 4000 and 4500. This difference cannot be easily explained. Indeed, even if the distribution of the maintenance costs difference ΔC_m seems more concentrated in the negative part, i.e. that C_m for the fleet method is higher than for the 1VS1 method, no specific pattern can be identified (Figure 7.17). The delay costs are in many scenarios higher for the 1VS1 method than for the fleet method.

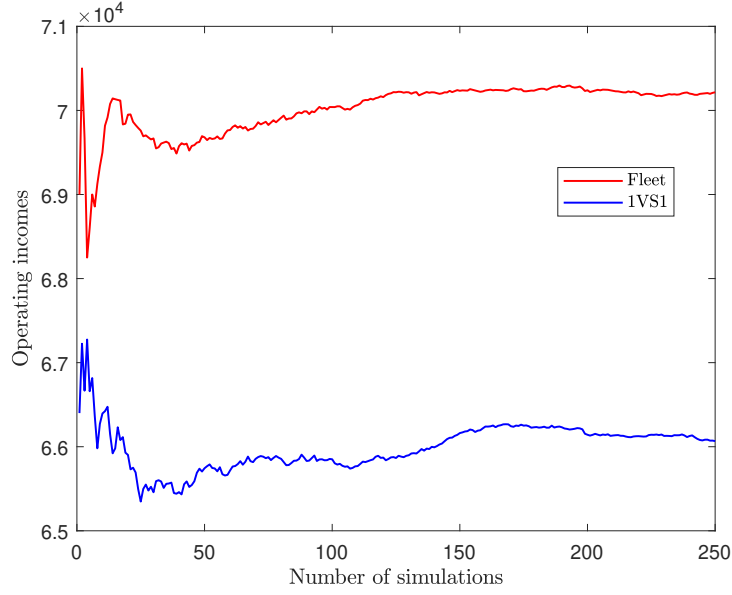


Figure 7.15: Operating incomes convergence for the fleet and the 1VS1 methods

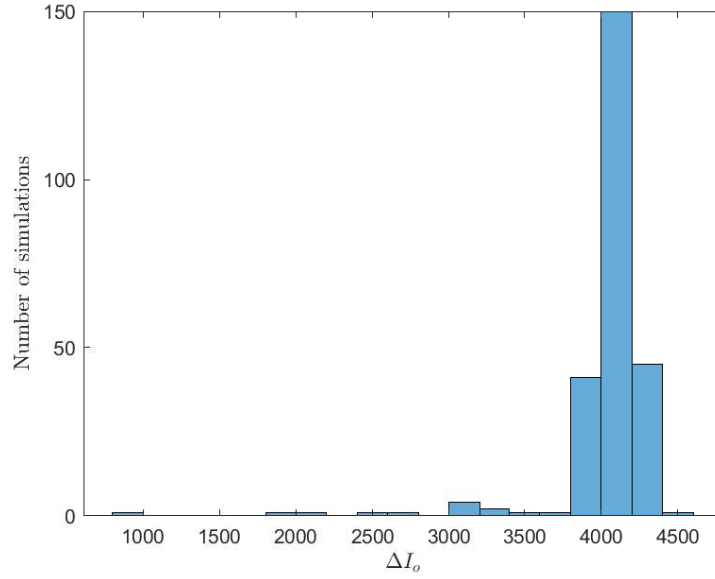


Figure 7.16: Distribution of the operating incomes difference ΔI_o between the two methods for the simulations

The average number of rescheduling n_r is smaller for the 1VS1 method than for the fleet method with values respectively equal to 3.95 and 4.58 (Figure 7.18). It is due to the fact that there are many more preventive maintenance operations competed in the schedules obtained with the 1VS1 method. It also justify why n_d and n_f are in average smaller for the 1VS1 method. With more blocks, the risk of failure is smaller, so n_f is smaller. Moreover,

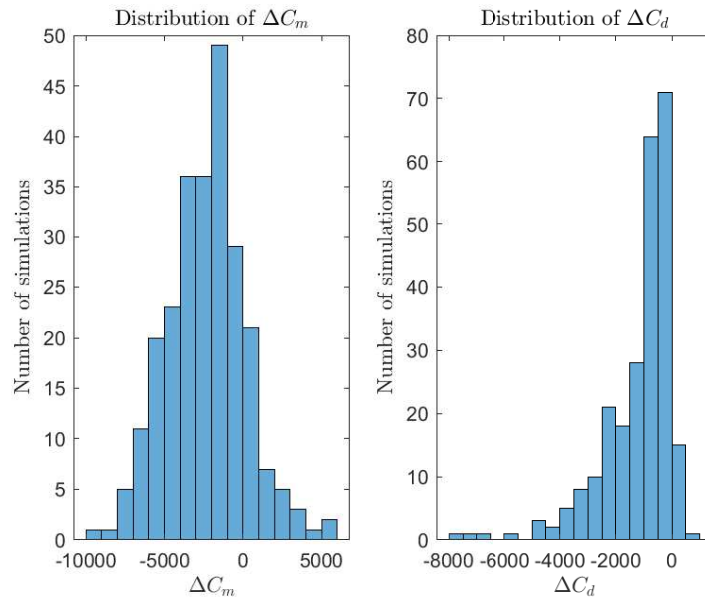


Figure 7.17: Distribution of the maintenance costs and delay costs difference denoted ΔC_m and ΔC_d between the two methods for the simulations

the deterioration information may not bring enough value to trigger a rescheduling so n_d is smaller for the 1VS1 method than for the fleet method.

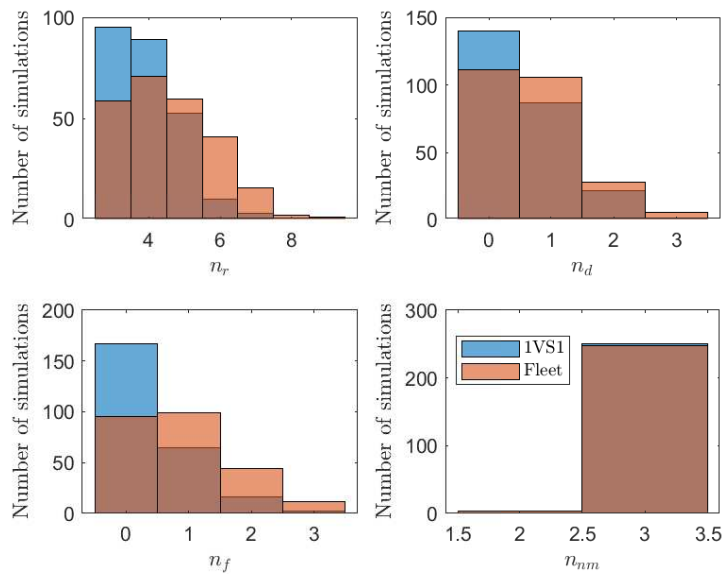


Figure 7.18: Distributions of the total number of rescheduling n_r and the numbers of rescheduling due to deterioration information, failure and new missions respectively denoted n_d , n_f and n_{nm}

7.6 Conclusion

In this chapter, we have studied the joint scheduling problem for a fleet of vehicles. In a first step, we consider the problem as static and have adapted the genetic algorithm developed for the static joint scheduling for a single vehicle to integrate the fleet dimension and coordinate the decision-making processes enabling to dispatch the missions between the different vehicles of the fleet and schedule at the suitable time the maintenance operations to avoid failures and delays in the missions completion.

We notice with the different application example that applying a centralized approach that makes decisions for the whole fleet by considering the information available on every vehicle is more profitable than a decentralized approach that avoid any synchronization between the vehicles activities. This effect is further highlighted when we consider an heterogeneous fleet with vehicles having different configurations. As there are no monitoring information considered in this study case, the information mentioned previously are the missions features and the vehicle configurations.

From these promising results, we study the dynamic joint scheduling problem for a fleet. In this case, we take into account failure occurrences, available deterioration measurements and new requested missions as opportunities to update the initial schedule for the fleet. The first results on specific study cases show that synchronizing all the available information coming from all the vehicles enable to make better decision and decrease the maintenance costs as well as the risk of delay. Considering the fleet enable to swap the missions between the different vehicles if one of them has a too high failure risk that could jeopardize the proper conduct of the mission. We should however remain careful about rescheduling insofar as rescheduling too often could be less profitable from a logistic point of view. Indeed, if rescheduling implies swapping too much the missions between the different vehicles there is a significant cost to consider to load and unload the goods from one vehicle to another.

Nevertheless, the results show that the development of a fleet management solution based on a joint optimization of the maintenance planning and the missions scheduling offer huge opportunities to improve the customers profitability. It could become a great decision support tool to increase their operating incomes while using at its maximum the capacity of the fleet.

Conclusions and perspectives

Contents

8.1	Conclusions	199
8.2	Perspectives	201
8.2.1	Short term perspectives	201
8.2.2	Long term perspectives	202

8.1 Conclusions

The research work presented in this thesis studies the joint scheduling problem for missions and maintenance operations for either a single industrial vehicle or for a fleet of vehicles. We focus on three main axes: how to jointly schedule the missions and the maintenance operations for a deteriorating vehicle, how to design a dynamic method to integrate the monitoring information to update the joint schedule and how to integrate the fleet dimension to improve the decision-making process if we have to manage the activities of more than one vehicle. At the moment, the maintenance planning and the mission scheduling are handled completely independently. The maintenance planning is defined by the truck manufacturer and based on the vehicle configuration and the intended usage conditions. The maintenance operations are scheduled at regular intervals for each component independently. The fleet management part in the sense of missions assignment to the vehicles is handled by the truck owner. When it comes to fleet management, the truck manufacturer proposes some services to enable the customer to evaluate the state of its fleet by it is not a tool to schedule the missions to complete.

The objective of this PhD is to develop a methodology linking the maintenance operations scheduling and the fleet management aspect i.e. the mission scheduling and evaluate the added value that such a methodology could bring for the customer to help him make decisions to manage its fleet of trucks. The long-term goal for the Volvo Group is to increase its service offer. A fleet management service designed as a decision support tool for the customer is seen as valuable to improve the customer profitability and satisfaction. To develop a methodology that dynamically defines a joint schedule for the missions and the maintenance operations for a fleet of deteriorating vehicles, we have to split the work into different stages to consider each complexity level at a time.

- **Static joint scheduling of missions and maintenance for a deteriorating vehicle**

The first idea is to study the static joint scheduling problem for a single vehicle. This joint scheduling problem links to different activities: the operating activity when the vehicles is completing missions and the maintenance activity. To obtain a relevant schedule, it needs to be adapted to the vehicle usage conditions and to the evolution of its health state. A deterioration model for the vehicle has then been defined to consider the impact of missions having different severity of usage and different operating conditions. Based on this model, it is possible to optimize the maintenance costs by grouping the missions into different blocks separated by maintenance operations. The grouping decisions are directly related to the remaining useful lifetime of the vehicle. The developed methodology is based on a genetic algorithm to optimize the maintenance costs. It is a first step towards our final objective but the main issue is that the schedule is static and cannot evolve over time according to the available monitoring information and the events that will occur during the schedule execution.

- **Dynamic joint scheduling of missions and maintenance for a deteriorating vehicle**

To upgrade the methodology, we study how to integrate the real time information to improve the decision-making process and update the schedule to adapt to the real health state of the vehicle. We consider three different potential information to update the schedule. The first one is the failure occurrence. Naturally, such an occurrence has to trigger a rescheduling because the current schedule does not fit to the vehicle deterioration state anymore. The second one is the new missions that can be requested during the schedule execution. These missions may be top priority ones so they absolutely need to be added to the schedule as soon as possible to avoid any delay on the deliveries.

- **Static and dynamic joint scheduling of missions and maintenance for a fleet of deteriorating vehicles**

Most of the Volvo Group customers own fleets of vehicles. They are seeking services to help them managing at best their fleet to optimize their profit. Considering the joint scheduling problem from a fleet perspective is then the next step to pursue. Based on the previous work, integrating the fleet dimension into the optimized strategy to jointly schedule the missions and the maintenance operations adds a new level of complexity and multiplies the possibility to schedule and reschedule the missions. The first part of the work deals with the static joint scheduling problem to identify the potential added value brought by the integration of the fleet dimension in the developed methodology. When considering an heterogeneous fleet of vehicles, the schedules obtained with the fleet based optimization method significantly increase the global operating incomes for the fleet.

This approach has then been improved to take into account the monitoring information, the failure occurrences and the new requested missions to complete. This centralized approach

enables to make decision based on the global state of the fleet and tends to make better decisions than a decentralized approach that schedules independently the activities for each vehicle. Through application example, we manage to emphasize the added value brought by such an approach to dynamically define a joint schedule for the missions and the maintenance operations for a fleet of deteriorating vehicles.

These conclusions show that the contributions of the thesis enable to meet the industrial objectives and methodological challenges initially defined. This thesis work has also opened several research perspectives that we discuss now.

8.2 Perspectives

8.2.1 Short term perspectives

- **Pursuing the development of the dynamic joint scheduling methodology for a fleet of vehicles**

The short term perspectives should focus on pursuing the development of the methodology to solve the dynamic joint scheduling problem for a fleet of deteriorating vehicles.

The first step is to go on with the performance analysis to draw some general conclusions and perform sensitivity study to analyse the influence of the decision parameters on both the quality and robustness of the obtained schedule for the fleet and on the global operating incomes earned with this schedule. Then, the methodology should also be compared with other strategies such as centralized load balancing strategies based on decision threshold [149].

Finally, we should also check the performance of the methodology to solve larger size problems for which monitoring information are more often available. It will probably lead to improve the strategy to trigger rescheduling and to define whether or not a rescheduling should concern the whole fleet or just a part of the fleet.

- **Removing the simplifying assumptions**

A work to remove the different hypotheses we made should be carried out. These main hypotheses are about the deterioration process and the vehicle modelling.

We assumed that we know perfectly the parameters of the deterioration process according to the different operating conditions met during the missions as well as according to the type of vehicle we use to complete it. It would be relevant to analyse the evolution of the maintenance costs and the operating incomes if these parameters are not exactly known.

We also decided to model the vehicle as a single-component system. That is why, it is characterized by a global health indicator. Many research work have been made to optimize

the maintenance planning for multi-component systems to adjust to the system use and the users constraints [31] and to ensure that the vehicle is available and performing well on a determined period [97]. A work to incorporate some parts regarding the way to model the system should be considered.

- **Integrating the geographical dimension in the decision-making process**

There are also other aspects that should be investigated when scheduling the maintenance activities. The capacity of the workshops and the availability of the resources to carry out the maintenance operations in good conditions would be interesting subjects. These aspects directly leads to a major aspect that we do not take into account in this thesis. It is the geographical dimension. Indeed, we schedule both the missions and the maintenance operations without considering the location of the different delivery points or the location of the workshop to perform the maintenance operations. In this case, the missions and the maintenance actions scheduling depend not only on the vehicle deterioration evolution but also on the opportunities offered by its geographical location[122].

8.2.2 Long term perspectives

This thesis proposes a customized methodology to jointly schedule the missions and the maintenance operations on a fleet of deteriorating vehicles. The fleet can be composed of either the same range of vehicles or can be an heterogeneous fleet with different vehicle configurations available.

- **Implementing the developed methods on concrete fleet management cases**

As the Volvo Group business model is changing to go towards an enhanced development of connected services mostly about predictive maintenance and fleet management, it would be relevant to test this methodology in a real and concrete study case. It is naturally the most long term perspective insofar as different stakeholder would be required, and not only the Volvo Group. Indeed, the first step is to build a strong collaboration with a customer that could give us information about how he schedules the missions for his fleet and the available input parameters for the missions. Except some average usage trends, we have no knowledge whatsoever on how the vehicles are really used. The second step is to secure the access to the monitoring data regarding the components health state. We need the right data with the right quality to make good decisions. If we manage to meet all these conditions, a pilot project could be implemented to evaluate on a concrete case the added value of the developed joint scheduling methodology for the missions and the maintenance operations.

- **Towards the development of a prescriptive maintenance approach**

The expectation of customers is no longer only to have an alert of abnormal failure or degradation, but rather to have access to a set of solutions and recommendations regarding maintenance, but also in terms of usage and operating methods, to keep the vehicle as operational as possible. The Volvo Group aims at developing post-prognostic decision-making methods to help reinforce the shift from predicting deterioration and remaining useful lifetime of a system to prescribing optimal actions on the system, in terms of mode of use or maintenance, in a prescriptive maintenance approach [107, 41]. The objective is to propose a complete decision-making process based on the connected data regarding the vehicle health state, its usage conditions, its spatial location and its operating constraints. For instance, depending on the state of health of the vehicle and its current mission, the kind of decision to make could be if it is more appropriate to go to the workshop to repair the damaged components before the breakdown or to modify the vehicle configuration and parametrization to adapt its operating parameters to its current use and thus complete the mission. It is no longer just a question of maintenance scheduling according to the vehicle deterioration, but also of incorporating the possibility of changing the vehicle settings so that it can continue its on going mission or wait for the best time and place to carry out its repair. The post-prognosis decision and the resulting prescriptions may therefore concern both maintenance and/or a change in use, or even an adaptation of the monitoring strategy when the state of health is too deteriorated.

Bibliography

- [1] Abdel-Hameed M. “A Gamma Wear Process.” In: *IEEE Transactions on Reliability* 24.2 (1975), pp. 152–153 (Citation on page 34).
- [2] AFNOR. *Terminologie de la maintenance [In French]*. NF-EN-13306-X-60-319. 2001 (Citations on pages 40, 41, and 42).
- [3] AFNOR. *Terminologie relative à la fiabilité - Maintenabilité - Disponibilité [In French]*. X NF, 60-500. 1988 (Citation on page 25).
- [4] Aggoune R. “Minimizing the makespan for the flow shop scheduling problem with availability constraints.” In: *European Journal of Operational Research* 153.3 (2004), pp. 534–543 (Citations on pages 74 and 85).
- [5] Aggoune R. and Portmann M.C. “Flow shop scheduling problem with limited machine availability: A heuristic approach.” In: *International Journal of Production Economics* 99.1-2 (2006), pp. 4–15 (Citation on page 74).
- [6] Ait Kadi D. and Cléroux R. “Replacement strategies with mixed corrective actions at failure.” In: *Computers & Operations Research* 18.2 (1991), pp. 141–149 (Citations on pages 63 and 64).
- [7] Allaoui H. and Artiba A. “Scheduling two-stage hybrid flow shop with availability constraints.” In: *Computers & Operations Research* 33.5 (2006), pp. 1399–1419 (Citation on page 74).
- [8] Allaoui H., Lamouri S., Artiba A., and Aghezzaf E. “Simultaneously scheduling n jobs and the preventive maintenance on the two-machine flow shop to minimize the makespan.” In: *International Journal of Production Economics* 112.1 (2008), pp. 161–167 (Citation on page 73).
- [9] Anbar D. “An asymptotically optimal inspection policy.” In: *Naval Research Logistics Quarterly* 23.2 (1976), pp. 211–218 (Citation on page 64).
- [10] Andréasson N. “Optimization of opportunistic replacement activities in deterministic and stochastic multi-component systems.” Master’s thesis. Chalmers University of Technology, Sweden, 2004 (Citations on pages 61 and 66).
- [11] Antelman G.R. and Savage R.I. “Surveillance problems: Wiener processes.” In: *Naval Research Logistics Quarterly* 12.1 (1965), pp. 35–55 (Citation on page 34).
- [12] Archibald Y.W. and Dekker R. “Modified block-replacement for multiple-component systems.” In: *IEEE Transactions on Reliability* 45.1 (1996), pp. 75–83 (Citation on page 63).
- [13] Asmussen S. *Applied Probability and Queues*. Vol. 51. Application of Mathematics. Springer-Verlag, New York, 2003 (Citation on page 34).
- [14] Assaf D. and Shanthikumar J.G. “Optimal Group Maintenance Policies with Continuous and Periodic Inspections.” In: *Management Science* 33.11 (1987), pp. 1440–1452 (Citation on page 62).

- [15] Aytug H., Lawley M.A., McKay K., Mohan S., and Uzsoy R. “Executing production schedules in the face of uncertainties: A review and some future directions.” In: *European Journal of Operational Research* 161.1 (2005), pp. 86–110 (Citation on page 82).
- [16] Bagdonavicius V. and Nikulin M. “Estimation in Degradation Models with Explanatory Variables.” In: *Lifetime Data Analysis* 7 (Mar. 2001), pp. 85–103 (Citation on page 36).
- [17] Bajestani M.A. “Integrating Maintenance Planning and Production Scheduling: Making Operational Decisions with a Strategic Perspective.” PhD thesis. University of Toronto, 2014 (Citation on page 71).
- [18] Barlow R. and Hunter L.C. “Optimum Preventive Maintenance Policies.” In: *Operations Research* 8.1 (1960), pp. 90–100 (Citation on page 54).
- [19] Barlow R.E., Hunter L.C., and Proschan F. “Optimum Checking Procedures.” In: *Journal of the Society for Industrial and Applied Mathematics* 11.4 (1963), pp. 1078–1095 (Citation on page 56).
- [20] Barlow R.E. and Proschan F. *Mathematical Theory of Reliability*. John Wiley & Sons, Incorporated, 1996 (Citations on pages 25, 43, 51, and 54).
- [21] Barros A. “Maintenance des systèmes multicomposants sous surveillance imparfaite : modélisation stochastique et optimisation [In French].” PhD thesis. Université de Technologie de Troyes, France, 2006 (Citations on pages 47 and 61).
- [22] Benbouzid-Sitayeb F., Bessadi Y., Guebli S.A., Varnier C., and Zerhouni N. “Résolution du problème de l’ordonnancement conjoint maintenance/production par la stratégie séquentielle.” In: *4e Conférence Francophone de Modélisation et de Simulation, Toulouse, France*. Vol. 2. 2003, pp. 627–633 (Citation on page 72).
- [23] Benbouzid-Sitayeb F., Varnier C., Zerhouni N., Guebli S.A., and Bessadi Y. “Joint scheduling of jobs and preventive maintenance operations in the flowshop sequencing problem: A resolution with sequential and integrated strategies.” In: *International Journal of Manufacturing Research* 6.1 (2011), pp. 30–48 (Citations on pages 71, 78, and 85).
- [24] Bencheikh G., Letouzey A., and Desforges X. “Process for joint scheduling based on health assessment of technical resources.” In: *10th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes SAFEPROCESS*. Vol. 51. 24. Warsaw, Poland, 2018, pp. 192–199 (Citation on page 86).
- [25] Berg M. and Epstein B. “A modified block replacement policy.” In: *Naval Research Logistics Quarterly* 23.1 (1976), pp. 15–24 (Citations on pages 56 and 63).
- [26] Berrichi A. and Yalaoui F. “Efficient bi-objective ant colony approach to minimize total tardiness and system unavailability for a parallel machine scheduling problem.” In: *The International Journal of Advanced Manufacturing Technology* 68.9-12 (2013), pp. 2295–2310 (Citation on page 79).

- [27] Berrichi A., Yalaoui F., Amodeo L., and Mezghiche M. “Bi-Objective Ant Colony Optimization approach to optimize production and maintenance scheduling.” In: *Computers & Operations Research* 37.9 (2010), pp. 1584–1596 (Citations on pages 79 and 86).
- [28] Blazewicz J., Beit J., Formanowicz P., Kubiak W., and Schmidt G. “Heuristic algorithms for the two-machine owshop with limited machine availability.” In: *Omega* 29 (2001), pp. 599–608 (Citation on page 73).
- [29] Bouche M., Plauchu V., and Retour D. “Les Systèmes Experts Appliqués à la Maintenance Modifient-ils la Donne ?” In: *Systèmes Experts en Maintenance [In French]*. Masson, Paris. 1987 (Citation on page 40).
- [30] Boulenger A. *Vers le Zéro Panne avec la Maintenance Conditionnelle*. AFNOR, Paris, 1988 (Citation on page 58).
- [31] Bouvard K. “Planification de la maintenance basée sur la dégradation de composants. Application à un véhicule industriel. [In French].” PhD thesis. Université de Lille 1: Sciences et Technologies, France, 2010 (Citations on pages 6, 13, 29, 31, 34, 37, 45, 50, 51, 61, and 202).
- [32] Bouvard K., Artus S., Bérenguer C., and Cocquempot V. “Condition-based dynamic maintenance operations planning & grouping. Application to commercial heavy vehicles.” In: *Reliability Engineering & System Safety* 96.6 (2011), pp. 601–610 (Citations on pages 45 and 66).
- [33] Bérenguer C., Grall A., Dieulle L., and Roussignol M. “Maintenance policy for a continuously monitored deteriorating system.” In: *Probability in the Engineering and Informational Sciences* 17 (2003), pp. 235–250 (Citation on page 48).
- [34] Cadini F., Zio E., and Avram D. “Monte Carlo-based filtering for fatigue crack growth estimation.” In: *Probabilistic Engineering Mechanics* 24 (2009), 367–373 (Citation on page 47).
- [35] Cassady C.R. and Kutanoglu E. “Integrating Preventive Maintenance Planning and Production Scheduling for a Single Machine.” In: *IEEE Trans. Rel.* 54.2 (2005), pp. 304–309 (Citations on pages 19, 20, 75, and 76).
- [36] Castanier B. “Modélisation stochastique et optimisation de la maintenance conditionnelle des systèmes à dégradation graduelle [In French].” PhD thesis. Université de Technologie de Troyes, France, 2002 (Citation on page 32).
- [37] Castanier B., Grall A., and Bérenguer C. “A condition-based maintenance policy with non-periodic inspections for a two-unit series system.” In: *Reliability Engineering & System Safety* 87 (2005), pp. 109–120 (Citation on page 65).
- [38] Chen W.J. “Minimizing number of tardy jobs on a single machine subject to periodic maintenance.” In: *Omega* 37.3 (2009), pp. 591–599 (Citation on page 72).
- [39] Cho D. and Parlar M. “A survey of maintenance models for multi-unit systems.” In: *European Journal of Operational Research* 51.1 (1991), pp. 1–23 (Citations on pages 61 and 66).

- [40] Christer A.H. and Wang W. “A model of condition monitoring of a production plant.” In: *International Journal of Production Research* 30.9 (1992), pp. 2199–2211 (Citations on pages 47, 48, and 58).
- [41] Clausen U., Langkau S., and Kreuz F. “Using the Digital Shadow for a Prescriptive Optimization of Maintenance and Operation.” In: *Advances in Production, Logistics and Traffic*. Springer, Cham, 2019, pp. 265–276 (Citation on page 203).
- [42] Coffman E.G. Jr., Garey M.R., and Johnson D.S. “Approximation algorithms for bin packing: A survey.” In: *Approximation Algorithms for NP-Hard Problems* (1996), pp. 46–93 (Citation on page 108).
- [43] French National Road Committee. *Long Haul 40T*. 2018. URL: <http://www.cnr.fr/en/Indices-Statistics/Long-haul-artic/CNR-indices#haut> (Citation on page 3).
- [44] Coudert T., Grabot B., and Archimède B. “Logique floue et système multi-agents pour un ordonnancement coopératif production/maintenance.” In: *3e Conférence Franco-phone de Modélisation et de SIMulation (MOSIM)*. Vol. 1. 2001, pp. 485–490 (Citation on page 87).
- [45] Cox D.R. “Regression Models and Life Table.” In: *Journal of the Royal Statistical Society. Series B* 34 (1972), pp. 187–202 (Citation on page 31).
- [46] Da W., Feng H., and Pan E. “Integrated preventive maintenance and production scheduling optimization on uniform parallel machines with deterioration effect.” In: *IEEE International Conference on Industrial Engineering and Engineering Management*. 2016, pp. 951–955 (Citations on pages 19 and 81).
- [47] Dekker R. and Scarf P. “On the Impact of Optimisation Models in Maintenance Decision Making: The State of the Art.” In: *Reliability Engineering & System Safety* 60 (1997), pp. 111–119 (Citation on page 61).
- [48] Dekker R., Smit A., and Losekoot J. “Combining maintenance activities in an operational Planning Phase: a Set-Partitioning approach.” In: *IMA Journal of Management Mathematics* 3.4 (1991), pp. 315–331 (Citation on page 66).
- [49] Deloux E. “Politiques de maintenance conditionnelle pour un système à dégradation continue soumis à un environnement stressant [In French].” PhD thesis. Université de Nantes, France, 2008 (Citations on pages 31 and 32).
- [50] Deloux E., Castanier B., and Bérenguer C. “Maintenance policy for a deteriorating system evolving in a stressful environment.” In: *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability* 222.4 (2008), pp. 613–622 (Citation on page 36).
- [51] Dieulle L., Bérenguer C., Grall A., and Roussignol M. “Sequential condition-based maintenance scheduling for a deteriorating system.” In: *European Journal of Operational Research* 150 (2003), pp. 451–461 (Citation on page 47).
- [52] Do Van P., Barros A., Bérenguer C., Bouvard K., and Brissaud F. “Dynamic grouping maintenance with time limited opportunities.” In: *Reliability Engineering & System Safety* 120 (2013), pp. 51–59 (Citation on page 66).

- [53] Do Van P., Vu H.C., Barros A., and Bérenguer C. “Grouping maintenance strategy with availability constraint under limited repairmen.” In: vol. 8. Mexico, 2012 (Citation on page 66).
- [54] Endrenyi J., Aboresheid S., Allan R.N., Anders G.J., Asgarpour S., Billinton R., Chowdhury N., Dialynas E.N., Fipper M., Fletcher R.H., Grigg C., McCalley J., Meliopoulos S., Mielnik T.C., Nitu P., Rau N., Reppen N.D., Salvaderi L., Schneider A., and Singh C. “The present status of maintenance strategies and the impact of maintenance on reliability.” In: *IEEE Transactions on Power Systems* 16.4 (2001), pp. 638–646 (Citation on page 44).
- [55] Feldman R.M. “Optimal Replacement with Semi-Markov Shock Models.” In: *Journal of Applied Probability* 13.1 (1976), pp. 108–117 (Citation on page 32).
- [56] Qiang Feng, Xiong Bi, Xiujie Zhao, Yiran Chen, and Bo Sun. “Heuristic hybrid game approach for fleet condition-based maintenance planning.” In: *Reliability Engineering & System Safety* 157 (2017), pp. 166–176 (Citation on page 87).
- [57] Feng H., Da W., and Xi L. “Joint optimization of flowshop sequence-dependent manufacturing cell scheduling and preventive maintenance.” In: *IEEE International Conference on Industrial Engineering and Engineering Management*. 2016, pp. 946–950 (Citations on pages 19, 20, and 80).
- [58] Feng Q., Li S., and Sun B. “A multi-agent based intelligent configuration method for aircraft fleet maintenance personnel.” In: *Chinese Journal of Aeronautics* 27.2 (2014), pp. 280–290 (Citation on page 87).
- [59] Fitouhi M.C. “Optimisation de la planification intégrée de la maintenance préventive et de la production des systèmes multi-états [In French].” PhD thesis. Université Laval, Québec, 2011 (Citations on pages 83, 84, 85, and 86).
- [60] Fitouhi M.C. and Noureldath M. “Integrating noncyclical preventive maintenance scheduling and production planning for a single machine.” In: *International Journal of Production Economics* 136.2 (2012), pp. 344–351 (Citation on page 76).
- [61] Fitouhi M.C. and Noureldath M. “Integrating noncyclical preventive maintenance scheduling and production planning for multi-state systems.” In: *Reliability Engineering & System Safety* 121 (2014), pp. 175–186 (Citation on page 81).
- [62] Floudas C.A. and Gounaris C.E. “A review of recent advances in global optimization.” In: *Journal of Global Optimization* 45.1 (2009), pp. 3–38 (Citations on pages 83 and 84).
- [63] Fouladirad M. and Grall A. “Condition-based maintenance for a system subject to a non-homogeneous wear process with a wear rate transition.” In: *Reliability Engineering & System Safety* 96.6 (2011), pp. 611–618 (Citation on page 49).
- [64] Fouladirad M., Grall A., and Dieulle L. “On the use of on-line detection for maintenance of gradually deteriorating systems.” In: *Reliability Engineering & System Safety* 93 (2008), pp. 1814–1820 (Citation on page 36).

- [65] Fusaro R.E., Nielsen J.P., and Scheike T.H. “Marker-dependent hazard estimation: An application to AIDS.” In: *Statistics in medicine* 12 (1993), pp. 843–865 (Citation on page 31).
- [66] Gertsbakh I. *Reliability Theory with Applications to Preventive Maintenance*. Springer. 2000 (Citation on page 51).
- [67] Ghelam S., Simeu-Abazi Z., Derain J.P., Feuillebois C., Vallet S., and Glade M. “Integration of health monitoring in the avionics maintenance system.” In: *IFAC Proceedings Volumes* 39.13 (2006), pp. 1449–1454 (Citation on page 29).
- [68] Gholami M., Zandieh M., and Alem-Tabriz A. “Scheduling hybrid flow shop with sequence-dependent setup times and machines with random breakdowns.” In: *The International Journal of Advanced Manufacturing Technology* 42.1-2 (2009), pp. 189–201 (Citation on page 74).
- [69] Ghomi E.J., Rahmani A.M., and Qader N.N. “Load-balancing algorithms in cloud computing: A survey.” In: *Journal of Network and Computer Applications* 88 (2017), pp. 50–71 (Citation on page 164).
- [70] Goffaux G., Vande Wouwer A., and Bernard O. “Improving continuous—discrete interval observers application to microalgae-based bioprocesses.” In: *Journal of Process Control* 19 (2009), pp. 1182–1190 (Citation on page 48).
- [71] Grall A., Bérenguer C., and Dieulle L. “A condition-based maintenance policy for stochastically deteriorating systems.” In: *Reliability Engineering & System Safety* 76.2 (2002), pp. 167–180 (Citation on page 48).
- [72] Volvo Group. *Technology Plan - An overview: 2025 and beyond*. 2018 (Citation on page 11).
- [73] Hadidi L.A., Al-Turki U.M., and Rahim A. “Integrated models in production planning and scheduling, maintenance and quality: a review.” In: *International Journal of Industrial and Systems Engineering* 10.1 (2012), pp. 21–50 (Citations on pages 19, 71, and 72).
- [74] Ho J.C. and Wong J.S. “Makespan Minimization for m Parallel Identical Processors.” In: *Naval Research Logistics* 42.6 (1995), pp. 935–948 (Citation on page 73).
- [75] Holland J.H. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. Cambridge, MA, USA: MIT Press, 1992 (Citation on page 106).
- [76] Hoogeveen H., Lenté C., and T’kindt V. “Rescheduling for new orders on a single machine with setup times.” In: *European Journal of Operational Research* 223.1 (2012), pp. 40–46 (Citation on page 83).
- [77] Huggare J. “An autonomous future takes shape.” In: *Volvo Group Magazine* (2019), pp. 18–19 (Citation on page 11).
- [78] Huynh K. “Quantification de l’apport de l’information de surveillance dans la prise de décision en maintenance. [In French].” PhD thesis. Université de Technologie de Troyes, France, 2011 (Citations on pages 43, 45, and 47).

- [79] Jansen J. and Van der Duyn Schouten F. “Maintenance optimization on parallel production units.” In: *IMA Journal of Mathematics Applied in Business and Industry* 6 (1995), pp. 113–134 (Citation on page 62).
- [80] Khoury E. “Modélisation de la durée de vie résiduelle et maintenance prédictive : application à des véhicules industriels. [In French].” PhD thesis. Université de Technologie de Troyes, France, 2012 (Citations on pages 30, 45, 47, 50, 60, and 99).
- [81] Khoury E., Deloux E., Grall A., and Bérenguer C. “An adaptive sequential maintenance decision for a deteriorating system with covariates and maintenance constraints.” In: *Proc Reliability, Risk and Safety (ESREL 2011)*. 2011, pp. 2271–2278 (Citations on pages 48 and 104).
- [82] Kiani A. and Taghipour S. “Joint optimization of maintenance and production scheduling.” In: *Reliability and Maintainability Symposium (RAMS)*. IEEE, 2017, pp. 1–6 (Citation on page 76).
- [83] Kubiak W., Blazewicz J., Formanowicz P., Beit J., and Schmidt G. “Two-machine flow shops with limited machine availability.” In: *European Journal of Operational Research* 136 (2002), pp. 528–540 (Citation on page 73).
- [84] Kubzin M.A. and Strusevich V.A. “Two-machine flow shop no-wait scheduling with machine maintenance.” In: *A Quarterly Journal of Operations Research* 3.4 (2005), pp. 303–313 (Citation on page 73).
- [85] Kumar U., Knezevic J., and Crocker J. “Maintenance free operating period – an alternative measure to MTBF and failure rate for specifying reliability?” In: *Reliability Engineering & System Safety* 64 (1999), pp. 127–131 (Citation on page 66).
- [86] Kuo Y. and Chang Z.A. “Integrated production scheduling and preventive maintenance planning for a single machine under a cumulative damage failure process.” In: *Naval Research Logistics* 54.6 (2007), pp. 602–614 (Citation on page 75).
- [87] Ladj A., Benbouzid-Si Tayeb F., and Varnier C. “An integrated prognostic based hybrid genetic-immune algorithm for scheduling jobs and predictive maintenance.” In: *2016 IEEE Congress on Evolutionary Computation (CEC)*. Vancouver, BC, Canada: IEEE, 2016, pp. 2083–2089 (Citations on pages 19, 20, 78, 111, and 112).
- [88] Laggoune R., Chateauneuf A., and Aissani D. “Opportunistic policy for optimal preventive maintenance of a multi-component system in continuous operating units.” In: *Computers & Chemical Engineering* 33 (2009), pp. 1499–1510 (Citation on page 65).
- [89] Laprie J.C. *Guide de la Sûreté de Fonctionnement [In French]*. Cépaduès. 1996 (Citation on page 24).
- [90] Lawless J. and Crowder M. “Covariates and Random Effects in a Gamma Process Model with Application to Degradation and Failure.” In: *Lifetime Data Analysis* 10.3 (2004), pp. 213–227 (Citation on page 36).
- [91] L’Ecuyer P. and Haurie A. “Preventive Replacement for Multicomponent Systems: An Opportunistic Discrete-Time Dynamic Programming Model.” In: *IEEE Transactions on Reliability* 32.1 (1983), pp. 117–118 (Citation on page 61).

- [92] Lee C.Y. “Minimizing the makespan in the two-machine flowshop scheduling problem with an availability constraint.” In: *Operations Research Letters* 20.3 (1997), pp. 129–139 (Citation on page 71).
- [93] Lee C.Y. “Parallel machines scheduling with nonsimultaneous machine available time.” In: *Discrete Applied Mathematics* 30.1 (1991), pp. 53–61 (Citation on page 73).
- [94] Lee C.Y. and Chen Z.L. “Scheduling jobs and maintenance activities on parallel machines.” In: *Naval Research Logistics* 47.2 (2000), pp. 145–165 (Citation on page 73).
- [95] Lee C.Y. and Liman S.D. “Capacitated two-parallel machines scheduling to minimize sum of job completion times.” In: *Discrete Applied Mathematics* 41.3 (1993), pp. 211–222 (Citation on page 73).
- [96] Lee W.C. and Wu C.C. “Multi-machine scheduling with deteriorating jobs and scheduled maintenance.” In: *Applied Mathematical Modelling* 32.3 (2008), pp. 362–373 (Citation on page 71).
- [97] Lesobre R. “Modélisation et optimisation de la maintenance et de la surveillance des systèmes multi-composants - Applications à la maintenance et à la conception de véhicules industriels. [In French].” PhD thesis. Université Grenoble Alpes, France, 2015 (Citations on pages 4, 7, 14, 16, 18, 31, 32, 33, 34, 37, 46, 50, 51, 61, 66, 97, 100, 102, 104, and 202).
- [98] Lesobre R., Bouvard K., Bérenguer C., Barros A., and Cocquempot V. “A Usage-Informed Preventive Maintenance Policy to Optimize the Maintenance Free Operating Period for Multi-Component Systems.” In: *Probabilistic Safety Assessment and Management (PSAM 12)*. Honolulu, 2014 (Citations on pages 45 and 66).
- [99] Lesobre R., Bouvard K., Bérenguer C., and Grall A. “Tailored maintenance optimization within Maintenance on Demand project.” In: *Proceedings of ESREL 2012*. Helsinki, Finland, 2012 (Citation on page 13).
- [100] Li H., Li M., Liu Q., and Li S. “Integrated optimization research on preventive maintenance planning and production scheduling.” In: *Management and Service Science (MASS), 2010 International Conference on*. IEEE, 2010, pp. 1–5 (Citation on page 76).
- [101] Liao C.J. and Chen W.J. “Single-machine scheduling with periodic maintenance and nonresumable jobs.” In: *Computers & Operations Research* 30.9 (2003), pp. 1335–1347 (Citation on page 72).
- [102] Liao C.J., Shyur D.L., and Lin C.H. “Makespan minimization for two parallel machines with an availability constraint.” In: *European Journal of Operational Research* 160.2 (2005), pp. 445–456 (Citation on page 73).
- [103] Lin C.H. and Liao C.J. “Makespan minimization for two parallel machines with an unavailable period on each machine.” In: *The International Journal of Advanced Manufacturing Technology* 33 (2007), pp. 1024–1030 (Citation on page 73).
- [104] Liao H., Elsayed A.E., and Chan L.Y. “Maintenance of continuously monitored degrading systems.” In: *European Journal of Operational Research* 175.2 (2006), pp. 821–835 (Citation on page 45).

- [105] Liao L.M. and Tsai C.H. “Heuristic algorithms for two-machine flowshop with availability constraints.” In: *Computers & Industrial Engineering* 56.1 (2009), pp. 306–311 (Citation on page 73).
- [106] Liao W., Zhang X., and Jiang M. “An optimization model integrated production scheduling and preventive maintenance for group production.” In: IEEE, 2016, pp. 936–940 (Citation on page 77).
- [107] Liu B., Lin J., Zhang L., and Kumar U. “A Dynamic Prescriptive Maintenance Model Considering System Aging and Degradation.” In: *IEEE Access* 7 (2019), pp. 94931–94943 (Citation on page 203).
- [108] Low C., Hsu C.J., and Su C.T. “Minimizing the makespan with an availability constraint on a single machine under simple linear deterioration.” In: *Computers & Mathematics with Applications* 56.1 (2008), pp. 257–265 (Citation on page 72).
- [109] Low C., Ji M., Hsu C.J., and Su C.T. “Minimizing the makespan in a single machine scheduling problems with flexible and periodic maintenance.” In: *Applied Mathematical Modelling* 34.2 (2010), pp. 334–342 (Citations on pages 71 and 72).
- [110] “L’Union européenne s’accorde sur une réduction de 30% des émissions de CO₂ des camions [In French].” In: *La Tribune* (2018) (Citation on page 2).
- [111] Marseguerra M., Zio E., and Podofilini L. “Condition-based maintenance optimization by means of genetic algorithms and Monte Carlo simulation.” In: *Reliability Engineering & System Safety* 77 (2002), pp. 151–165 (Citation on page 47).
- [112] Masduff A. “Digitalization.” In: *Volvo Group Magazine* (2017), pp. 8–15 (Citation on page 11).
- [113] Michalewicz Z. *Genetic Algorithms + data Structures = Evolution Programs*. Springer series Artificial Intelligence. Berlin ; New York: Springer, 1996 (Citations on pages 88, 109, and 111).
- [114] MIL-HDBK-217F. *Reliability Prediction of Electronic Equipment*. U.S. Department of Defense, Washington, DC. 1991 (Citation on page 28).
- [115] *MoDe: Maintenance on Demand*. 2012. URL: https://fp7-mode.eu/?About_MoDE (Citation on page 12).
- [116] Mosheiov G. “Minimizing the sum of job completion times on capacitated parallel machines.” In: *Mathematical and Computer Modelling* 20.6 (1994), pp. 91–99 (Citation on page 73).
- [117] Mosheiov G. “Multi-Machine Scheduling With Linear Deterioration.” In: *INFOR* 36 (1998), pp. 205–214 (Citation on page 71).
- [118] Myers L.E. “Survival functions induced by stochastic covariate processes.” In: *Journal of Applied Probability* 18.2 (1981), 523–529 (Citation on page 31).
- [119] Nahas N. “Optimisation de la performance de systèmes multi-composants assujettis à des défaillances aléatoires [In French].” PhD thesis. Université Laval, Québec, 2008 (Citations on pages 83 and 84).

- [120] Nakagawa T. “Optimal policy of continuous and discrete replacement with minimal repair at failure.” In: *Naval Research Logistics Quarterly* 31.4 (1984), pp. 543–550 (Citation on page 54).
- [121] Nelson W. *Accelerated Testing: Statistical Models, Test Plans and Data Analysis*. Wiley Series in Probability and Statistics. Wiley, New York, 1990 (Citation on page 28).
- [122] Nguyen H.S.H., Do P., Vu H.C., and Iung B. “Dynamic maintenance grouping and routing for geographically dispersed production systems.” In: *Reliability Engineering & System Safety* 185 (2019), pp. 392–404 (Citation on page 202).
- [123] Nicolai R.P. “Maintenance Models for Systems subject to Measurable Deterioration.” PhD thesis. Erasmus Universiteit Rotterdam, 2008 (Citations on pages 28, 29, 31, and 62).
- [124] Nicolai R.P. and Dekker R. “Optimal Maintenance of Multi-component Systems: A Review.” In: *Complex System Maintenance Handbook*. Springer London, 2008, pp. 263–286 (Citation on page 62).
- [125] Nicolai R.P., Frenk H., and Dekker R. “Modeling and optimizing imperfect maintenance of coatings on steel structures.” In: *Structural Safety* 31.3 (2007), pp. 234–244 (Citation on page 44).
- [126] Nourelfath M., Fitouhi M.C., and Machani M. “An Integrated Model for Production and Preventive Maintenance Planning in Multi-State Systems.” In: *IEEE Transactions on Reliability* 59.3 (2010), pp. 496–506 (Citations on pages 79 and 81).
- [127] Okumoto K. and Elsayed E.A. “An optimum group maintenance policy.” In: *Naval Research Logistics Quarterly* 30.4 (1983), pp. 667–674 (Citation on page 62).
- [128] OREDA. *Offshore Reliability Data*. 4th ed. 2002 (Citation on page 28).
- [129] Ouali M.S., Ait Kadi D., and Gharbi A. “A simulation model for opportunistic maintenance strategies.” In: *7th IEEE International Conference on Emerging Technologies and Factory Automation*. Vol. 1. 1999, pp. 703–708 (Citation on page 65).
- [130] Ovalle Villamil, A., Hably A., Bacha S., Ramos G., and Hossain J.M. “Escort Evolutionary Game Dynamics Approach for Integral Load Management of Electric Vehicle Fleets.” In: *IEEE Transactions on Industrial Electronics* 64.2 (2017), pp. 1358–1369 (Citation on page 164).
- [131] Park K.S. “Optimal continuous-wear limit replacement under periodic inspections.” In: *IEEE Transactions on Reliability* 37.1 (1988), pp. 97–102 (Citation on page 48).
- [132] Petchrompo S. and Parlikad A.K. “A review of asset management literature on multi-asset systems.” In: *Reliability Engineering & System Safety* 181 (2019), pp. 181–201 (Citations on pages 20 and 81).
- [133] Pham H. and Wang H. “Imperfect maintenance.” In: *European Journal of Operational Research* 94.3 (1996), pp. 425–438 (Citation on page 54).
- [134] Pierrat L. and D’Aubigny G. “Somme de lois gamma différentes et détermination des paramètres d’une distribution gamma équivalente [In French].” In: *36eme Journées de Statistique*. SFdS. Montpellier, France, 2004 (Citation on page 100).

- [135] Pinedo M.L. *Scheduling: Theory, Algorithms, and Systems*. Third Edition. Springer Publishing Company, Incorporated, 2008 (Citation on page 70).
- [136] *Preventive Maintenance Intervals*. Service Bulletin Trucks. Renault Trucks, 2018 (Citation on page 6).
- [137] Rausand M. and Høyland A. *System reliability theory: models, statistical methods, and applications*. 2nd ed. Wiley series in probability and statistics. Hoboken, NJ: Wiley-Interscience, 2004 (Citations on pages 25, 29, 30, 41, and 52).
- [138] Raza S.A., Al-Turki U.M., and Selim S.Z. “Early-Tardy Minimization for Joint Scheduling of Jobs and Maintenance Operations on a Single Machine.” In: *International Journal of Operations Research* 4.1 (2007), pp. 32–41 (Citation on page 72).
- [139] Robert E., Bouvard K., Lesobre R., and Bérenguer C. “Joint assignment of missions and maintenance operations for a fleet of deteriorating vehicles.” In: *11th International Conference on Mathematical Methods in Reliability - MMR2019*. Ed. by M. Xie. Proc. of 11th International Conference on Mathematical Methods in Reliability - MMR2019. City University Hong-Kong. Hong-Kong, China, 2019, Paper #202 (Citation on page 169).
- [140] Robert E., Bérenguer C., Bouvard K., Tedie H., and Lesobre R. “A predictive approach to jointly schedule missions and maintenances for a deteriorating vehicle.” In: *28th European Safety and Reliability Conference - ESREL 2018*. Ed. by S. Haugen, A. Barros, C. van Gulijk, T. Kongsvik, and J.E. Vinnem. Safety and Reliability – Safe Societies in a Changing World - Proc. of 28th European Safety and Reliability Conference. NTNU-ESRA. Trondheim, Norway: Taylor & Francis (CRC Press/Balkema), 2018, pp. 529–537 (Citation on page 94).
- [141] Robert E., Bérenguer C., Bouvard K., Tedie H., and Lesobre R. “Dynamic sequential decision making for missions and maintenances scheduling for a deteriorating vehicle.” In: *PSAM14 - 14th International Probabilistic Safety Assessment and Management Conference*. Ed. by Johnson D. Droguett E.L. Proceedings of PSAM14 - 14th International Probabilistic Safety Assessment and Management Conference. Los Angeles, United States, 2018, Paper #31, 12p. (Citation on page 132).
- [142] Robert E., Bérenguer C., Bouvard K., Tedie H., and Lesobre R. “Joint dynamic scheduling of missions and maintenance for a commercial heavy vehicle: value of on-line information.” In: *IFAC-PapersOnLine*. Vol. 51. Warsaw, Poland, 2018, pp. 837–842 (Citations on pages 132 and 144).
- [143] Robinson A.M. “The trends shaping cities.” In: *Volvo Group Magazine* (2017), pp. 8–15 (Citation on page 10).
- [144] Rohlfshagen P. and Bullinaria J.A. “Nature inspired genetic algorithms for hard packing problems.” In: *Annals of Operations Research* 179.1 (2010), pp. 393–419 (Citation on page 109).
- [145] Ross S.M. “A model in which component failure rates depend on the working set.” In: *Naval Research Logistics Quarterly* 31.2 (1984), pp. 297–300 (Citation on page 61).
- [146] Ross S.M. *Stochastic processes*. 2nd edition. New York: John Wiley & Sons, Inc, 1996 (Citation on page 34).

- [147] Sadfi C., Penz B., Rapine C., Błażewicz J., and Formanowicz P. “An improved approximation algorithm for the single machine total completion time scheduling problem with availability constraints.” In: *European Journal of Operational Research* 161.1 (2005), pp. 3–10 (Citation on page 72).
- [148] Sandström J. “The economics of an electric switch.” In: *Volvo Group Magazine* (2019), pp. 16–17 (Citation on page 12).
- [149] Saxena A.B. and Sharma D. “Analysis of threshold based centralized load balancing policy for heterogeneous machines.” In: *International Journal of Advanced Information Technology* 1 (2011), p. 16 (Citation on page 201).
- [150] Scarf P.A. “A Framework for Condition Monitoring and Condition Based Maintenance.” In: *Quality Technology & Quantitative Management* 4.2 (2007), pp. 301–312 (Citation on page 44).
- [151] Scheaffer R.L. “Optimum Age Replacement Policies with an Increasing Cost Factor.” In: *Technometrics* 13.1 (1971), pp. 139–144 (Citation on page 54).
- [152] Schmidt G. “Scheduling Independent Tasks with Deadlines on Semi-identical Processors.” In: *Journal of the Operational Research Society* 39.3 (1988), pp. 271–277 (Citation on page 73).
- [153] Schmidt G. “Scheduling with limited machine availability.” In: *European Journal of Operational Research* 121 (2000), pp. 1–15 (Citation on page 72).
- [154] Sherif Y.S. and Smith M.L. “Optimal Maintenance Models for Systems Subject to Failure — A Review.” In: *Naval Research Logistics Quarterly* 28 (1981), pp. 47–74 (Citation on page 51).
- [155] Singpurwalla N. “Survival in Dynamic Environments.” In: *Statistical Science* 10.1 (1995), pp. 86–103 (Citation on page 30).
- [156] Sivanandam S.N. and Deepa S.N. *Introduction to Genetic Algorithms*. 1st. Berlin, New York: Springer Publishing Company, Incorporated, 2007 (Citations on pages 88, 90, 91, 106, and 171).
- [157] Sortrakul N., Nachtmann H.L., and Cassady C.R. “Genetic algorithms for integrated preventive maintenance planning and production scheduling for a single machine.” In: *Computers in Industry* 56.2 (2005), pp. 161–168 (Citation on page 76).
- [158] Sriram C. and Haghani A. “An optimization model for aircraft maintenance scheduling and re-assignment.” In: *Transportation Research Part A: Policy and Practice* 37.1 (2003), pp. 29–48 (Citation on page 81).
- [159] Tango T. “Extended block replacement policy with used items.” In: *Journal of Applied Probability* 15.3 (1978), 560–572 (Citation on page 56).
- [160] “The importance of transport.” In: *Volvo Group Magazine* (2018), pp. 30–31 (Citation on page 2).
- [161] Thomas, L.C. “A Survey of Maintenance and Replacement Models for Maintainability and Reliability of Multi-Item Systems.” In: *Reliability Engineering* 16.4 (1986), pp. 297–309 (Citation on page 61).

- [162] Tijms H.C. “Stochastic Modelling and Analysis: A Computational Approach.” In: (1986) (Citation on page 46).
- [163] Renault Trucks. *Optifleet [In French]*. 2019. URL: <https://www.renault-trucks.fr/optifleet> (Citation on page 9).
- [164] Volvo Trucks. *Fleet Management System - Dynafleet*. 2019. URL: <https://www.volvotrucks.co.uk/en-gb/services/dynafleet.html> (Citation on page 8).
- [165] Tsang A.H.C. “Strategic dimensions of maintenance management.” In: *Journal of Quality in Maintenance Engineering* 8.1 (2002), pp. 7–39 (Citation on page 42).
- [166] Valdez-Flores C. and Feldman R. “A Survey of Preventive Maintenance Models for Stochastically Deteriorating Single-Unit System.” In: *Naval Research Logistics (NRL)* 36 (1989), pp. 419–446 (Citations on pages 51, 54, and 57).
- [167] Van Dijkhuizen G.C. and Van Harten A. “Optimal clustering of frequency-constrained maintenance jobs with shared set-ups.” In: *European Journal of Operational Research* 99 (1997), pp. 552–564 (Citation on page 63).
- [168] Van Noortwijk J.M. “A survey of the application of gamma processes in maintenance.” In: *Reliability Engineering & System Safety* 94.1 (2009), pp. 2–21 (Citations on pages 34, 35, and 98).
- [169] Vaurio J.K. “Availability and cost functions for periodically inspected preventively maintained units.” In: *Reliability Engineering & System Safety* 63 (1999), pp. 133–140 (Citation on page 64).
- [170] Vieira G.E., Herrmann J.W., and Lin E. “Rescheduling manufacturing systems: a framework of strategies, policies, and methods.” In: *Journal of Scheduling* 6.1 (2003), pp. 39–62 (Citations on pages 20, 82, 133, and 160).
- [171] Vu H.C., Do Van P., Barros A., and Bérenguer C. “Maintenance activities planning and grouping for complex structure systems.” In: *Annual European Safety and Reliability Conference* (2012) (Citation on page 66).
- [172] Vu H.C., Do Van P., Barros A., and Bérenguer C. “Maintenance planning and dynamic grouping for multi-component systems with positive and negative economic dependencies.” In: *IMA Journal of Management Mathematics* 26.2 (2015), pp. 145–170 (Citation on page 66).
- [173] Wang D.J., Liu F., Wang J.J., and Wang Y.Z. “Integrated rescheduling and preventive maintenance for arrival of new jobs through evolutionary multi-objective optimization.” In: *Soft Computing* 20.4 (2016), pp. 1635–1652 (Citations on pages 20 and 83).
- [174] Wang H. “A survey of maintenance policies of deteriorating systems.” In: *European Journal of Operational Research* 139 (2002), pp. 469–489 (Citation on page 51).
- [175] Wang L., Chu J., and Mao W. “A condition-based replacement and spare provisioning policy for deteriorating systems with uncertain deterioration to failure.” In: *European Journal of Operational Research* 194 (2009), pp. 184–205 (Citation on page 49).
- [176] Wang S. and Liu M. “A branch and bound algorithm for single-machine production scheduling integrated with preventive maintenance planning.” In: *International Journal of Production Research* 51.3 (2013), pp. 847–868 (Citation on page 71).

- [177] Wang W. “A model to determine the optimal critical level and the monitoring intervals in condition-based maintenance.” In: *International Journal of Production Research* 38 (2000), pp. 1425–1436 (Citation on page 47).
- [178] Wang Y. and Pham A. “Imperfect preventive maintenance policies for two-process cumulative damage model of degradation and random shocks.” In: *International Journal of System Assurance Engineering and Management* 2 (Mar. 2011), pp. 66–77 (Citation on page 36).
- [179] Waslander S.L., Inalhan G., and Tomlin C.J. “Decentralized optimization via Nash bargaining.” In: *Theory and algorithms for cooperative systems* 4 (2004), pp. 565–585 (Citation on page 164).
- [180] Whitmore G.A. “Estimating degradation by a Wiener diffusion process subject to measurement error.” In: *Lifetime data analysis* 1 (Feb. 1995), pp. 307–19 (Citation on page 34).
- [181] Wildeman R.E. “The Art of Grouping Maintenance.” PhD thesis. Erasmus Universiteit Rotterdam, 1996 (Citations on pages 62 and 66).
- [182] Xiao L., Song S., Chen X., and Coit D.W. “Joint optimization of production scheduling and machine group preventive maintenance.” In: *Reliability Engineering & System Safety* 146 (2016), pp. 68–78 (Citation on page 80).
- [183] Xu Z., Ji Y., and Zhou D. “Real-time Reliability Prediction for a Dynamic System Based on the Hidden Degradation Process Identification.” In: *IEEE Transactions on Reliability* 57 (2008), pp. 230–242 (Citation on page 47).
- [184] Yalaoui A., Chaabi K., and Yalaoui F. “Integrated production planning and preventive maintenance in deteriorating production systems.” In: *Information Sciences* 278 (2014), pp. 841–861 (Citation on page 79).
- [185] Yam R.C.M., Tse P.W., Li L., and Tu P. “Intelligent Predictive Decision Support System for Condition-Based Maintenance.” In: *International Journal of Advanced Manufacturing Technology* 17 (2001), pp. 383–391 (Citation on page 44).
- [186] Yang D., Wang H., Feng Q., Ren Y., Sun B., and Wang Z. “Fleet-level selective maintenance problem under a phased mission scheme with short breaks: A heuristic sequential game approach.” In: *Computers & Industrial Engineering* 119 (2018), pp. 404–415 (Citation on page 87).
- [187] Yang D.L., Hsu C.J., and Kuo W.H. “A two-machine flowshop scheduling problem with a separated maintenance constraint.” In: *Computers & Operations Research* 35.3 (2008), pp. 876–883 (Citation on page 73).
- [188] Yang S.J., Yang D.L., and Cheng T.C.E. “Single-machine due-window assignment and scheduling with job-dependent aging effects and deteriorating maintenance.” In: *Computers & Operations Research* 37.8 (2010), pp. 1510–1514 (Citation on page 77).
- [189] Yulan J., Zuhua J., and Wenrui H. “Multi-objective integrated optimization research on preventive maintenance planning and production scheduling for a single machine.” In: *The International Journal of Advanced Manufacturing Technology* 39.9-10 (2008), pp. 954–964 (Citation on page 77).

- [190] Zhao X., Fouladirad M., Bérenguer C., and Bordes L. “Condition-based inspection/replacement policies for non-monotone deteriorating systems with environmental covariates.” In: *Reliability Engineering & System Safety* 95 (Aug. 2010), pp. 921–934 (Citation on page [36](#)).

Résumé — Ces travaux de thèse traitent des problèmes de planification conjointe des opérations de maintenance et des missions pour des flottes de véhicules industriels. Le but est de développer une méthodologie permettant d'adapter la planification conjointe des maintenances et des missions en fonction de l'état de santé des véhicules mais également en fonction des caractéristiques des missions. Ces caractéristiques correspondent aux conditions de sévérité d'usage qui ont un impact important sur la dégradation du camion et doivent être prise en compte pour adapter au mieux la planification des opérations de maintenance en fonction de l'évolution de la dégradation. La mise en place d'une méthodologie d'aide à la décision pour gérer une flotte permettrait d'améliorer la productivité, de réduire les coûts de maintenance tout en utilisant au mieux la capacité de la flotte. Cependant, le problème de planification conjointe pour une flotte est un problème complexe à résoudre nécessitant de considérer trois dimensions. La première est de planifier conjointement les missions et les maintenances de façon statique. La seconde est d'intégrer, de façon dynamique, les informations de surveillance disponibles et les différents événements pouvant se produire pour mettre à jour le planning. La troisième dimension est la dimension flotte qui implique de gérer plusieurs véhicules en parallèle. La première étape consiste à planifier conjointement les activités de maintenance et les missions pour un camion dans un cas statique. On suppose alors qu'on connaît toutes les missions à planifier et qu'aucune information de surveillance n'est disponible. Pour cela, on définit un modèle de dégradation du véhicule afin d'estimer sa durée de vie résiduelle pour prendre les décisions. C'est un modèle avec des valeurs de paramètres variables car le véhicule évolue dans des conditions de sévérité d'usage différentes en fonction des missions. Il est le point central pour mettre en place un algorithme de planification afin d'éviter les risques de panne trop importants. La planification est naturellement optimisée en fonction d'un critère basé soit sur les coûts de maintenance, soit sur les revenus d'exploitation. Une fois cette méthodologie définie, elle est complétée pour intégrer les informations relatives à la dégradation du véhicule, les occurrences de panne et les nouvelles missions qui peuvent demandées. Une approche dynamique pour résoudre le problème de planification pour un véhicule a donc été développée. Si une panne survient, le planning est mis à jour car il n'est plus adapté à l'évolution de la dégradation réelle du véhicule. De même, lorsque qu'une nouvelle mission est disponible, une mise à jour est indispensable car l'ordre de priorité des missions, défini par leurs dates limites de livraison, doit être pris en compte au plus vite pour éviter les pénalités de retard. En revanche, une information de dégradation peut avoir une influence variable sur le planning en cours. Il s'agit donc d'étudier la robustesse du planning pour éviter de changer trop souvent l'ordre des missions et les dates de maintenance. La dernière étape consiste à intégrer la dimension flotte dans la prise de décision. Ce n'est donc plus juste une question d'ordre des missions et de planification des maintenances au bon moment mais il faut également décider à quelle mission est attribuée à quel véhicule. La prise de décision dépend alors de l'état de la flotte. Une analyse de l'impact de la prise en compte de la dimension flotte dans le cas statique, puis dans le cas dynamique est menée. Des résultats de simulation permettent d'illustrer les méthodes développées et de montrer leur intérêt et les gains en termes de coûts qu'elles engendrent.

Abstract — This thesis work deals with the problems of joint scheduling for maintenance operations and missions for industrial vehicle fleets. The aim is to develop a methodology to adapt the joint scheduling of maintenance and missions according to the vehicles health state but also according to the missions features. These features correspond to the conditions of usage severity that have a significant impact on the truck deterioration and must be taken into account to adapt at best the maintenance operations schedule according to the deterioration evolution. The implementation of a decision support methodology to manage a fleet would improve productivity and reduce the maintenance costs while making the most of the fleet capacity. However, the joint scheduling problem for a fleet is a complex problem to solve and three main dimensions has to be considered. The first one is to jointly schedule missions and maintenance operations in a static case. The second one is to integrate the available monitoring information and the different events that can occur to update the schedule and treat the problem in a dynamic way. The third dimension is the fleet dimension that involves managing several vehicles in parallel. The first step is to jointly schedule the maintenance activity and the missions for a truck in a static case. It is assumed that all the missions to be performed are known and that no monitoring information is available. To do this, a vehicle deterioration model is defined to estimate its remaining useful lifetime to make decisions. It is a model with varying parameters since the vehicle operates under different conditions of usage severity according to the missions. It is the central point for setting up a scheduling algorithm to avoid any excessive risk of failure. The scheduling process is naturally optimized according to a criterion based on either the maintenance costs or the operating incomes. Once this methodology has been defined, it must be completed to include information on the vehicle deterioration, failure occurrences and new missions that may be requested. A dynamic approach has then been developed to solve the scheduling problem for a vehicle. If a breakdown occurs, the schedule must be updated because it is no longer adapted to the evolution of the current vehicle deterioration. Likewise, when a new mission is available, an update is essential because the priority order of the missions, defined by their deadlines, must be considered as soon as possible to avoid delay penalties. On the other hand, deterioration information can have a varying influence on the current schedule. Then, the schedule robustness has to be studied to avoid changing the mission order and the maintenance dates too often. The last step is to integrate the fleet dimension in the decision-making process. It is no longer just a question of mission order and timing for maintenance operations, but also of deciding which vehicle is assigned to which mission. The decision-making process then depends on the whole fleet. An analysis of the impact of considering the fleet dimension in the static case and then in the dynamic case is led. Simulation results are used to illustrate the developed methods and aim at showing their interest and the cost savings they generate.

