



Approche d'ingénierie systèmes pour l'intégration de la fabricabilité dans le processus de conception des systèmes mécatroniques

Mohamed Borchani

► To cite this version:

Mohamed Borchani. Approche d'ingénierie systèmes pour l'intégration de la fabricabilité dans le processus de conception des systèmes mécatroniques. Génie mécanique [physics.class-ph]. Université Paris Saclay (COMUE); Université de Tunis (1958-1988), 2019. Français. NNT : 2019SACLC043 . tel-02191513

HAL Id: tel-02191513

<https://theses.hal.science/tel-02191513>

Submitted on 23 Jul 2019

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.

Approche d'ingénierie systèmes pour l'intégration de la fabricabilité dans le processus de conception des systèmes mécatroniques

Thèse de doctorat de l'Université Paris-Saclay et de l'université de Tunis
Préparée à CentraleSupélec

École doctorale n°573 approches interdisciplinaires, fondements,
applications et innovation (Interfaces)

Ecole Doctorale Sciences et Techniques de l'Ingénieur de l'ENSIT

Spécialité de doctorat : Sciences et technologies industrielles
Spécialité de doctorat : Génie mécanique

Thèse présentée et soutenue à Saint Ouen, le 03/07/2019, par

Mohamed Firas Borchani

Composition du Jury :

Nabil Anwer	Président
Professeur, Université Paris Sud (LURPA)	
Vincent Cheutet	Rapporteur
Professeur, INSA Lyon (DISP- EA4570)	
Nizar Aifaoui	Rapporteur
Professeur, Université de Monastir (LGM)	
Abdelfattah Mlika	Examineur
Professeur, ENSI Sousse (LMS)	
Olivia Penas	Examinatrice
Ingénieur de recherche, Supméca (Quartz EA 7393)	
Jean-Yves Choley	Directeur de thèse
Professeur, Supméca (Quartz EA 7393)	
Noureddine Ben Yahia	Co-directeur de thèse
Professeur, ENSI Tunis (LMPE)	
Moncef Hammadi	Co-encadrant
Maitre de conférences, Supméca (Quartz EA7393)	

Titre : Approche d'ingénierie systèmes pour l'intégration de la fabricabilité dans le processus de conception des systèmes mécatroniques

Mots clés : MBSE, SBCE, Ingénierie systèmes, SysML, fiabilité, fabricabilité

Résumé : l'originalité de notre travail consiste à développer une approche qui intègre l'ingénierie systèmes basée sur les modèles (MBSE) avec les principes de l'approche SBCE afin de réduire les boucles itératives entre les concepteurs inexpérimentés, les fabricants et les experts. D'autres parties prenantes participent également au développement du produit afin d'assurer environnement d'ingénierie simultanée. Cette approche offre un grand potentiel pour gérer les aspects de fabricabilité et de fiabilité, en raison de la capacité à développer des connaissances en parallèle, en particulier pour les technologies avancées et non mures. L'avantage d'utilisation des principes du SBCE est d'offrir des possibilités de communication entre les parties prenantes au début de la phase de conception. SysML est utilisé comme langage descriptif pour représenter les différentes phases de cette approche. Pour étudier la robustesse ces système, nous avons développé un modèle intitulé «Model Based for Robustness Analysis» (MBRA) qui a pour objectif de filtrer les solutions faibles ou coûteuses au fil du temps et d'assister l'ingénieur système lors de l'adoption d'une étude de compromis. Pour l'étude de la fabricabilité des systèmes complexes, nous avons développé un autre modèle intitulé « Model Based for Manufacturability analysis »

Summary: the originality of our work is to develop an approach that integrates model-based systems engineering (MBSE) with the principles of the SBCE approach to reduce iterative loops between inexperienced designers, manufacturers, and experts. Other stakeholders are also involved in product development to ensure simultaneous engineering environment. This approach offers great potential for managing manufacturability and reliability aspects, due to the ability to develop parallel knowledge, especially for advanced and non-mature technologies. The advantage of using SBCE principles is to provide opportunities for communication between stakeholders early in the design phase. SysML is used as a descriptive language to represent the different phases of this approach. To study the robustness of these systems, we have developed a model called "Model Based for Robustness Analysis" (MBRA) which aims to filter low or expensive solutions over time and to assist the system engineer when adopting a trade-offs study. For the assessment of the manufacturability of complex systems, we have developed another model entitled "Model Based for Manufacturability analysis"

Remerciements

Mes remerciements s'adressent tout d'abord à Mr. Moncef Hammadi, Mr. Noureddine Ben Yahia et Mr. Jean-Yves Choley qui ont encadré cette recherche et dont les précieux conseils m'ont permis de mener à bien ce travail avec une grande économie de temps et d'efficacité. Je ne peux que leurs exprimer mon immense gratitude.

Je tiens également à remercier tous les membres du jury qui ont accepté d'évaluer cette thèse.

Je tiens en outre à remercier mes professeurs, mes collègues et mes amis qui ont bien voulu m'aider par leurs observations, remarques ou échange de documents et enrichir mon travail.

Je n'oublie pas ma mère et mon père dont la noblesse d'esprit et de coeur a été un havre de paix dans les moments difficiles de mon parcours, ni mon frère qui m'a toujours soutenu.

Je voudrais enfin remercier Abir qui a toujours été au rendez-vous et dont le soutien et le support ont constamment été inconditionnels et sans faille.

Table des matières

Introduction générale.....	1
CHAPITRE 1 : Etat de l'art.....	9
1.1. Ingénierie système	10
1.1.1 Aperçu général	10
1.1.2 Les normes de l'ingénierie systèmes.....	12
1.2. L'ingénierie systèmes basée sur les modèles (MBSE)	13
1.2.1 Généralités	13
1.2.2 Modèles de développement des cycles de vie en ingénierie système	14
1.2.3 Présentation du langage SysML.....	16
1.2.4 Application de l'MBSE pour le développement des modèles de conception	21
1.3. Utilisation de l'approche Lean pour le développement des produits complexes.....	27
1.3.1 Généralités at applications.....	27
1.3.2 L'approche SBCE	30
1.3.3 Application du SBCE.....	33
CHAPITRE 2 : Etude comparative sur les méthodes intégrant la fabricabilité	37
2.1. Aperçu sur la fabricabilité.....	38
2.2. Les critères d'évaluation de la fabricabilité :.....	39
2.3. Application de ces critères dans les travaux basés sur la fabricabilité	40
2.3.1 Reconnaissance des entités géométriques	40
2.3.2 La visualisation et le retour d'information (feedback).....	43
2.3.3 Les directives de la conception pour la fabrication (DFM).....	48
2.4. Tableau comparatif sur les méthodes intégrant la fabricabilité	53
2.5. Contribution de notre travail pour l'étude de la fabricabilité.....	55
Chapitre 3 : Développement du modèle MBRA pour le choix de la meilleure architecture système	58
3.1. Introduction.....	59
3.2. L'approche proposée.....	60
3.3. Implémentation de l'algorithme	69
3.4. Etude de cas d'un compresseur électrique	71
3.5. Comparaison avec des travaux similaires.....	81
3.6. Discussion	82
Chapitre 4 : Développement du modèle MBMA pour l'étude de la fabricabilité.....	86
4.1. Introduction.....	87

4.2.	L'approche proposée.....	88
4.3.	Implémentation de l'algorithme	97
4.4.	Etude de cas d'un boîtier papillon.....	99
4.5.	Discussion	114
Conclusion générale		117

Liste des figures

Fig.1 : Les boucles itératives entre les différents spécialistes.....	5
Fig.2 : Approche basée sur les ensembles [57]	6
Fig.3 : Coût de modification du produit après la phase de fabrication [96]	7
Fig.1.1 : Les activités incluses dans les normes de l'ingénierie systèmes [15]	12
Fig.1.2 : Modèles de développement du cycle de vie : (a) Cascade, (B) Spiral, (c) en V [28]....	15
Fig.1.3 : Relation entre SysML et UML 2.0 [32]	17
Fig.1.4 : Présentation des neuf diagrammes de SysML [34]	18
Fig.1.5 : Le modèle Harmony-SE [28]	21
Fig.1.6 : Le modèle RUP SE [28].....	22
Fig.1.7 : La méthodologie Vitech [28]	23
Fig.1.8 : La méthode OOSEM [31]	24
Fig.1.9 : Le modèle de développement des produits de la NASA [40].....	24
Fig.1.10 : La méthodologie de conception des systèmes mécatroniques [41]	25
Fig.1.11 : L'approche par point (Point-Based Engineering) [49]	28
Fig.1.12 : L'approche Set-Based Design [52].....	30
Fig.1.13 : Exemple d'application de l'approche SBCE dans la société Toyota [55]	31
Fig.1.14 : les étapes du SBCE [57]	32
Fig.1.15 : Présentation d'un cas d'étude de la pompe à jet [61]	34
Fig.1.16 : Application du SBCE dans l'entreprise Rolls-Royce [63].....	35
Fig.2.1 : Représentation des surfaces moyennes et des graphes d'adjacence [73]	42
Fig.2.2 : L'analyse de la fabricabilité par la visualisation [78]	44
Fig.2.3 : La représentation des alternatives de conception générées par le système CONDENSE pour l'étude de la fabricabilité [80].....	46
Fig.2.4 : Les modules d'analyse de la fabricabilité du logiciel CNC-RP [81]	47
Fig.2.5 : Les composants de l'outil CAMA pour l'analyse de la fabricabilité [85]	50
Fig.2.6 : Algorithme basé sur les directives du (DFM) pour l'étude de la fabricabilité des pièces moulées par injection [86]	51
Fig.2.7 : Modèle d'analyse de la fabricabilité pour le processus de rectification [87].....	52
Fig.2.8 : Les types de flux échangés entre l'ingénieur système et les participants à la conception.....	56
Fig.3.1 : Le modèle MBRA.....	61
Fig.3.2 : Le diagramme de profil SysML développé pour l'étude de la fiabilité.....	63
Fig.3.3 :l'organigramme de l'analyse de la fiabilité.....	70

Fig.3.4 : le compresseur électrique [92].....	72
Fig.3.5 : Le diagramme d'exigences SysML du compresseur électrique	73
Fig.3.6 : Le diagramme d'activités SysML du compresseur électrique	74
Fig.3.7 : Matrice d'allocation SysML du compresseur électrique	75
Fig.3.8 : L'architecture physique du compresseur électrique intégrée dans le diagramme de définition de blocs SysML.....	76
Fig. 3.9 : Attributs des composants du moteur sans balais intégrés dans le diagramme de définition de blocs SysML.....	78
Fig.3.10 : le graphe initial généré par Gephi pour l'étude de la fiabilité.....	79
Fig.3.11 : Le diagramme radar généré par Python pour l'analyse des compromis.....	80
Fig.4.1 : Le modèle MBMA	89
Fig.4.2 : le diagramme de profil SysML développé pour l'étude de la fabricabilité.....	90
Fig.4.3 : Le graphe initial généré par Gephi pour l'étude de la fabricabilité.....	98
Fig.4.4 : L'organigramme de l'analyse des compromis	98
Fig.4.5 : le boîtier papillon électronique	100
Fig.4.6 : Diagramme d'exigences SysML du boîtier Papillon intégrant les valeurs du client ..	101
Fig.4.7 : Diagramme d'activité SysML intégrant l'architecture fonctionnelle du boîtier papillon	102
Fig.4.8 : Matrice d'allocation SysML intégrant le cas du boîtier papillon retenu.....	104
Fig.4.9 : Diagramme d'exigences SysML intégrant les exigences fonctionnelles du corps de boîtier papillon	105
Fig.4.10 : Prototype 3D du boîtier papillon	106
Fig.4.11 : Implémentation des contraintes de conception dans le diagramme de définition de blocs SysML	107
Fig.4.12 : Implémentation des Contraintes des machines-outils dans le diagramme de définition de blocs SysML.....	108
Fig.4.13 : Implémentation des critères d'allocation entre la pièce et la machine-outil dans le diagramme d'exigences SysML	109
Fig.4.14 : Matrice d'allocation SysML illustrant la cartographie de l'espace des solutions....	110
Fig.4.15 : Matrice de satisfaction SysML illustrant les filtres utilisés pour l'ensemble des processus de fabrication	112
Fig.4.16 : Extraction des attributs du diagramme de définition de blocs SysML	113

Liste des tableaux

Tab.1 : Tableau comparatif sur les méthodes intégrant la fabricabilité	55
Tab.2 : les attributs de valeurs clé.....	72
Tab.3 : Les critères d'évaluation de la gravité des moteurs.....	77
Tab.4 : Les modes de défaillance des moteurs	77
Tab.5 : Les causes racines de chaque mode de défaillance des moteurs	77
Tab.6 : Intégration des principes du SBCE dans le modèle MBMA.....	97
Tab.7 : Cas d'étude des boîtiers papillon considérés pour le choix de la meilleure solution	103
Tab.8 : Critère d'évaluation de l'état de surface des entités de fabrication.....	107
Tab.9 : Paramètres géométriques pour l'évaluation de la capacité de l'ensemble des processus de fabrication	111

Glossaire

MBSE : Model-Based Systems Engineering

SBCE : Set-Based Concurrent Engineering

AMDEC : Analyse des modes de défaillance et de leurs criticités

FTA : Fault tree analysis

IS : Ingénierie systèmes

IATF 16949 : International Automotive Task Force 16949

SYSML : System Modeling Language

UML : Unified modeling language

MOF : Meta-Object Facility

XMI : XML Metadata Interchange

GrahML : Graph modeling language

Introduction g n rale

Contexte

Les industries manufacturi res en g n ral et du secteur automobile en particulier sont confront es   des exigences plus s v res des clients, une plus grande personnalisation des produits, des lots de production plus petits et des changements brusques de la chaine d'approvisionnement. Par ailleurs, une nouvelle tendance de produits intelligents est en train de changer les souhaits des clients en facilitant la gestion des t ches quotidiennes. Ces produits intelligents se caract risent par une int gration croissante de la m canique avec l' lectronique et le traitement de l'information. Cette int gration se situe entre les composants (mat riel) et les fonctions pilot es par l'information (logiciel), ce qui donne des syst mes int gr s appel s syst mes m catroniques. Le d veloppement d'un nouveau produit complexe comme les syst mes m catroniques n'est pas une t che facile en raison des interd pendances qui existent entre ses sous-syst mes. Ceci implique d'int grer les contraintes m tiers en aval telles que la fabrication, la qualit  et la fiabilit  au plut t dans le processus de conception et de d veloppement pour  tre comp titif sur le march  tout en r duisant le co t.

Pour faire face   ces d fis, les fabricants doivent choisir et int grer des technologies qui les aident   s'adapter rapidement face aux changements de la demande du march  et    lever la qualit  du produit tout en optimisant l'utilisation de l' nergie et des ressources. Ces technologies constituent le noyau d'un syst me de fabrication  mergent appel  « Smart Manufacturing system » (SMS), qui maximise le flux et la r utilisation des donn es dans toute l'entreprise. Ces syst mes s'appuient sur l'utilisation des normes pour g rer le cycle de vie des produits et sur l'approche « Lean Product Development system » (LPDS) issue du constructeur automobile Toyota pour optimiser la conception et la fabrication des syst mes complexes tout en  liminant les t ches sans valeur ajout e. De nouvelles initiatives   l' chelle mondiale ont vu

le jour pour contribuer à l'intégration des normes et de l'approche Lean dans les systèmes intelligents de fabrication.

L'industrie 4.0 est un nouveau paradigme qui contient une stratégie technique pour la réalisation des SMS [1]. Son objectif est la création d'innovation pour avoir des produits et des processus de fabrication intelligents qui permettent de minimiser le coût et le temps de mise sur le marché en adoptant la philosophie Lean. En effet, cette philosophie implique l'intégration de toutes les parties de la chaîne d'approvisionnement inter et intra-organisationnelles pouvant fonctionner en synergie pour créer un système efficace et de haute qualité permettant de livrer des produits robustes dans les délais fixés par le client [2].

Enjeux globaux

De nombreux travaux de recherche, tel que [3], ont été réalisés pour comprendre les défis de l'industrie automobiles liés à l'intégration des contraintes métiers en phase amont du processus de conception. En effet, les entreprises du secteur automobile cherchent à implémenter en premier lieu la norme IATF 16949 qui définit les exigences du système de management de la qualité pour les produits liés à l'automobile. Elle représente la rationalisation des spécifications adoptées par les industries de l'automobile dans le monde entier (AVSQ'94 en Italie, EAQF'94 en France, VDA 6.1 en Allemagne et QS 9000 aux Etats-Unis) en une seule spécification technique. Selon [4], l'objectif d'IATF 16949 est d'aider les organisations à fournir des produits ou des services qui répondent aux exigences des clients dans le secteur de l'automobile et de favoriser l'amélioration continue, mettant l'accent sur la prévention des rebuts dans la chaîne d'approvisionnement. La norme fournit également des moyens pour consolider et communiquer des concepts dans le domaine de la gestion de la qualité qui ont été approuvés par un comité de représentants de l'industrie automobile ainsi que des organismes internationaux de normalisation.

En deuxi me lieu, l'industrie automobile tend au perfectionnement continu de son syst me de production en int grant des approches telles que l'ing nierie concourante et l'approche Lean pour minimiser le temps de cycle et pour r pondre aux appels d'offres avec des d lais courts. Ceci implique d'int grer les m tiers couvrant le cycle de vie des produits, d s les phases amont du processus de conception et de d veloppement, o  70% des co ts de production sont d termin s. Il est donc essentiel de prendre en compte les contraintes de fabrication, de qualit  et de fiabilit  d s les premi res  tapes du processus de conception, d'identifier les param tres influen ant les performances du produit et son co t, puis de les  valuer de fa on   obtenir un produit optimal.

En Troisi me lieu, l'int gration d'outils et des langages de programmation dans la ch ne num rique de gestion du cycle de vie des produits permet d'automatiser quelques t ches r p titives, d'assurer la tra abilit  du produit et d'utiliser les ressources de mani re optimale.

Un autre enjeu majeur de l'industrie automobile est de capitaliser les informations, les connaissances et les exp riences acquises sur les projets ant rieurs. Malheureusement, ces connaissances sont d tenues par un nombre limit s d'experts au sein de l'entreprise et ne peuvent pas  tre exploitées pour une utilisation future. Ceci engendre des retards consid rables pour les projets industriels   venir. De ce fait, les entreprises cherchent   int grer une approche d'ing nierie syst mes bas e sur le retour d'exp rience.

Probl matique et solution propos e

Dans les sections pr c dentes, nous avons mis en perspective les enjeux des industries manufacturi res en g n rale et du secteur automobile en particulier. Ces enjeux sont associ s   la n cessit  de capitaliser les connaissances li es   l'int gration des m tiers qui couvrent le cycle de vie du produit telles que la fabrication, la qualit  et la fiabilit  d s la phase amont de la conception en adoptant l'approche concourante et la philosophie Lean ce qui constitue le

contexte de notre travail de recherche. Les sections suivantes nous serviront, d'une part,   d crire la probl matique de recherche qui  merge de ces attentes, et d'autre part   exposer la solution propos e.

Probl matique abord e

Le d veloppement d'un produit complexe tels que les syst mes m catroniques comprenant le choix de la meilleure architecture syst me en premier lieu et l' tude de la fabricabilit  en deuxi me lieu prend beaucoup de temps et de ressources et son industrialisation est souvent retard e en raison des modifications de conception fr quentes. Par exemple, 41 500 tentatives de modification de la conception ont  t  effectu es au cours du processus de conception d'un syst me de capteurs complexe au cours de ses huit ann es de d veloppement [5].

Dans la phase pr liminaire de conception, les d tails du produit sont faibles   cause du manque de participation des diff rents acteurs de conception (les ing nieurs, les experts, les parties prenantes et les clients) et   cause de la d consid ration des contraintes m tiers telles que la fabrication, la qualit  et la fiabilit  lors du d veloppement des produits en avant-phase. Par cons quent, les d tails g om triques et les caract ristiques des composants peuvent n cessiter de nombreux changements de conception ainsi que des v rifications ult rieures de fabricabilit  avant la phase d'industrialisation. Ces changements de conception sont dus entre autre   l'adoption de l'approche traditionnelle par point (Point-Based Design) qui consiste   choisir une seule alternative de l'espace de solutions puis faire des modifications chaque fois que les ing nieurs m tiers rencontrent des probl mes dans leurs domaines d'analyse. Pour un produit complexe, faire une conception de composant qui satisfait toutes les parties prenantes devient plus complexe en raison des interd pendances entre les composants et les sous-syst mes. En effet, les concepteurs inexp riment s ou d butants avec peu ou sans connaissance de la conception pour la fabrication (DFM) et de la conception pour la fiabilit  (DFR) produiraient

une conception avec une géométrie arbitraire. Ceci conduit à une communication itérative entre les concepteurs débutants, les ingénieurs métiers, les experts et les clients pour évaluer la fabricabilité et la robustesse de la pièce. Ces boucles itératives retardent le développement des produits au stade de la conception initiale. La figure 1 montre les différentes boucles itératives entre les participants de la conception.

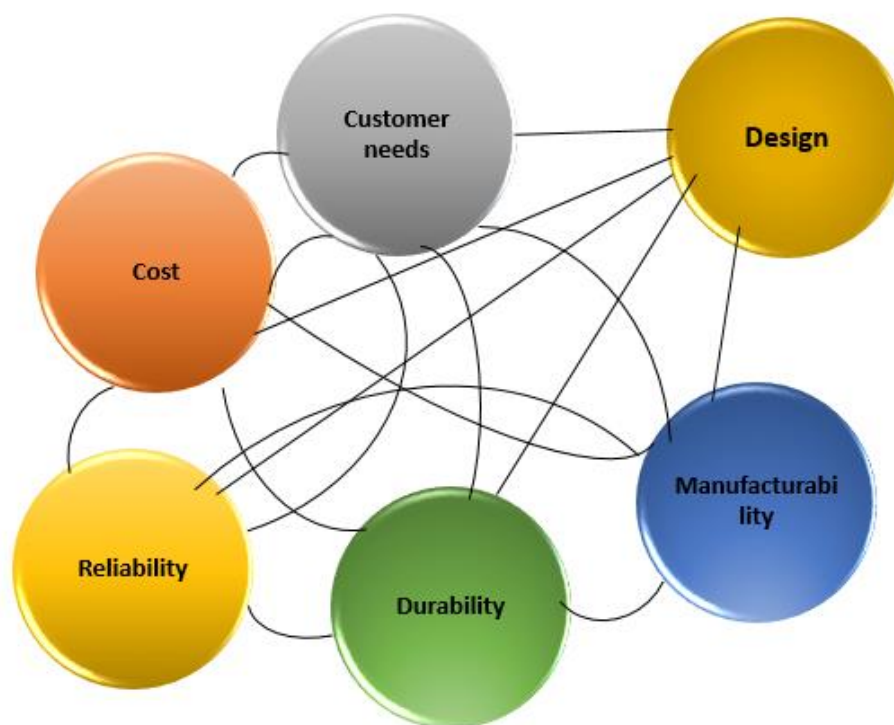


Fig.1 : Les boucles itératives entre les différents spécialistes

Ces boucles itératives ne peuvent pas être facilement retirées pour plusieurs raisons. Tout d'abord, l'adoption d'une approche par point conduit souvent à un conflit entre les ingénieurs métiers. Ensuite, les conceptions de pièces produites par des concepteurs ayant un niveau de connaissance de DFM et de DFR basique ou intermédiaire pour un processus de fabrication peuvent souvent être incomplètes et avoir une géométrie arbitraire. Ceci engendre la difficulté à formaliser de manière cohérente le savoir-faire métier nécessaire pour garantir un produit conforme aux attentes du client tout au long de son cycle de vie. Enfin, le processus de conception d'un produit complexe nécessite le développement d'un environnement d'ingénierie

simultanée dans lequel des équipes multidisciplinaires travaillent ensemble à partir des exigences des clients jusqu'au choix de la meilleure architecture système et du meilleur processus de fabrication qui optimise le coût, la qualité et le temps de cycle.

La solution proposée

Pour faire face aux constats remarqués précédemment, nous allons concentrer nos efforts sur l'intégration d'une approche d'ingénierie systèmes qui se base sur des modèles pour le choix de la meilleure architecture système et du meilleur processus de fabrication avec les principes du « Set-Based concurrent engineering » (SBCE). Le SBCE est une approche Lean issue du constructeur automobile Toyota et qui consiste à évaluer un large ensemble de solutions proposées par les différentes parties prenantes puis à sélectionner progressivement ces solutions, convergeant ensemble vers la solution répondant le mieux à "la voix du Client". La solution proposée est présentée dans la figure 2.

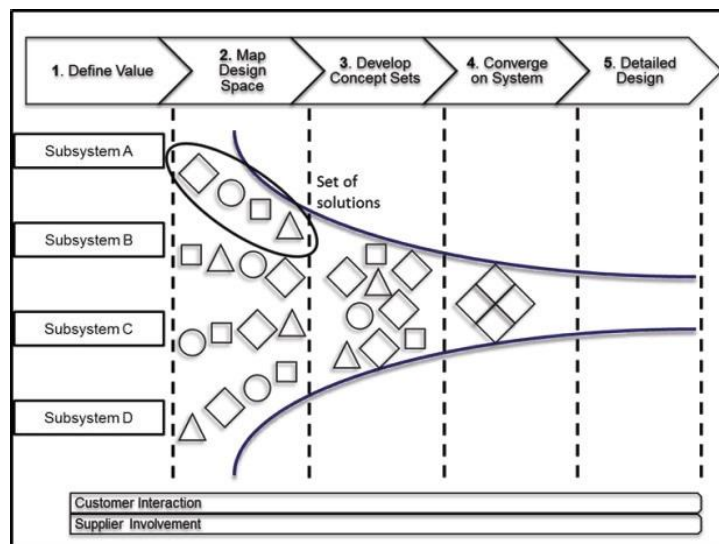


Fig.2 : Approche basée sur les ensembles [57]

Cette approche permet de réduire les boucles itératives en constituant un capital de connaissance et de savoir-faire et en tenant compte des contraintes de fabrication en phase de conception préliminaire ceci afin d'anticiper les problèmes, détectés traditionnellement dans

les phases situées en aval du cycle de vie du produit. Par conséquent, cette approche permet de générer de gains considérables en terme de coût et de temps.

Les méthodes existantes de conception nommées « Point Based Design» sont des méthodes itératives et n'interviennent qu'en phase de conception détaillée, avec le risque d'engendrer des coûts supplémentaires pour une modification quelconque.

La figure 3 montre l'impact de modification du produit après la phase de fabrication :

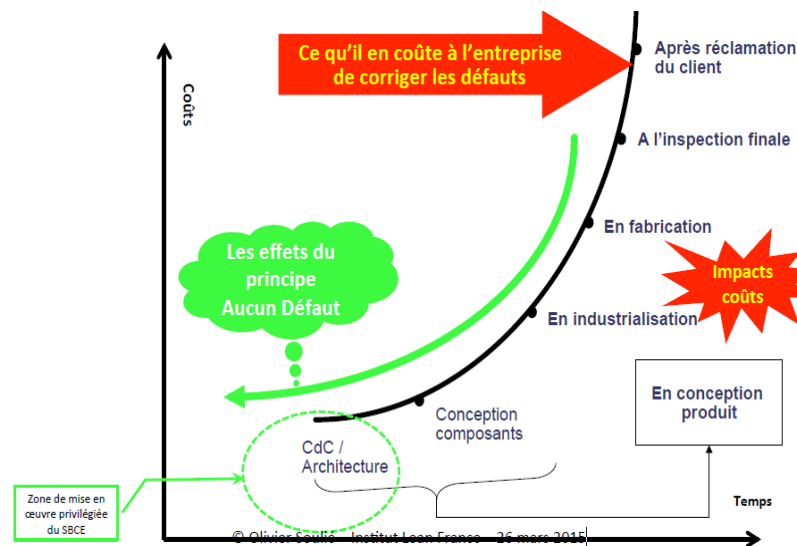


Fig.3 : Coût de modification du produit après la phase de fabrication [96]

L'intérêt actuel des entreprises, plus particulièrement dans le secteur automobile, est orienté vers l'intégration, au sein du processus de conception des produits, des contraintes de conception, de fabrication et de qualité dans un seul environnement qui seront partagées entre les parties prenantes afin de converger rapidement vers la meilleure solution.

Organisation du mémoire

Chapitre 1 – État de l'art. Ce chapitre présente les travaux de la littérature relatifs à la problématique proposée. Cette synthèse de l'état de l'art est axée sur les thèmes de l'ingénierie systèmes basée sur les modèles et l'utilisation des principes de l'approche Set-Based Concurrent Engineering afin de construire nos deux modèles.

Chapitre 2 – Etude comparative sur les méthodes intégrant la fabricabilité. Dans ce chapitre nous allons étudier quelques méthodes d’analyse de la fabricabilité et les critères d’évaluation dès la phase amont de conception et nous allons introduire notre approche basée sur les modèles.

Chapitre 3 – Développement du modèle MBRA pour le choix de la meilleure architecture système. Ce chapitre contient le premier modèle intitulé « Model-Based for Robustness Analysis » et développé à partir des diagrammes SysML et des principes du SBCE pour le choix de la meilleure architecture système. Un cas d’étude d’un compresseur électrique est proposé pour illustrer les différentes phases de ce modèle.

Chapitre 4 – Développement du modèle MBMA pour l’étude de la fabricabilité. Dans ce chapitre, nous allons présenter un nouveau modèle intitulé « Model-Based for Manufacturability Analysis » pour l’étude de la fabricabilité des pièces automobiles avant la phase d’industrialisation. Pour cela, nous allons étudier un boîtier papillon pour illustrer les phases de ce modèle.

CHAPITRE 1.

Etat de l'art

1.1.	Ingénierie système	10
1.1.1	Aperçu général	10
1.1.2	Les normes de l'ingénierie systèmes.....	12
1.2.	L'ingénierie systèmes basée sur les modèles (MBSE)	13
1.2.1	Généralités	13
1.2.2	Modèles de développement des cycles de vie en ingénierie système	14
1.2.3	Présentation du langage SysML.....	16
1.2.4	Application de l'MBSE pour le développement des modèles de conception	21
1.3.	Utilisation de l'approche Lean pour le développement des produits complexes.....	27
1.3.1	Généralités at applications.....	27
1.3.2	L'approche SBCE	30
1.3.3	Application du SBCE.....	33

1.1. Ingénierie système

1.1.1 *Aperçu général*

Plusieurs descriptions du mot système ont été utilisées dans la littérature. Parmi ces descriptions, nous citons les suivantes : « Un système est un ensemble de composants inter reliés qui interagissent les uns avec les autres d'une manière organisée pour accomplir une finalité commune » [6]. Une autre définition du mot système est la suivante : « Un système est un ensemble intégré d'éléments qui accomplissent un objectif défini » [7]. Dans le même contexte, l'ingénierie systèmes (IS) est définie en tant que démarche méthodologique pour maîtriser la conception des systèmes et produits complexes. Cette démarche permet aussi de faire évoluer et vérifier un système apportant une solution économique et performante aux besoins d'un client tout en satisfaisant l'ensemble des parties prenantes. L'ingénierie systèmes s'appuie sur des représentations à différents niveaux d'abstraction pour appréhender, conceptualiser, concevoir, estimer, simuler, valider, justifier des choix et communiquer [8].

Cette démarche joue un rôle important pour l'implication des différents acteurs métiers dans toutes les phases de développement du produit. En effet, dans le contexte de l'entreprise étendue actuelle telle que l'industrie automobile, les manufacturiers doivent faire face à des problèmes d'organisation liés aux contraintes d'acteurs délocalisés [9]. D'autres difficultés émergent également au niveau de l'intégration des contraintes de fabrication dès la phase amont de conception. Pour cela, l'adoption d'une approche d'ingénierie systèmes basée sur des modèles est une solution efficace pour diminuer le temps de cycle et pour améliorer la productivité.

L'ingénierie systèmes réunit :

- Les efforts techniques liés au développement, fabrication, vérification, déploiement, opérations, support, mise à disposition et formation d'utilisateurs des produits et processus d'un système.

- La définition et la gestion de la configuration du système.
- La traduction de la définition du système en terme d'une séquence de tâches structurées.
- Le développement d'informations pour la prise de décisions du management.

Le département de défense des Etats Unis (DOD) [10] définit l'ingénierie systèmes par : «
L'application d'efforts scientifiques et d'ingénierie pour :

- Transformer un besoin opérationnel en une description de paramètres de performance d'un système et une configuration du système à travers l'utilisation d'un processus itératif de définition, de synthèse, d'analyse, de conception, de test et d'évaluation.
- Intégrer des paramètres techniques reliés et s'assurer de la compatibilité des interfaces physiques, fonctionnelles et des programmes d'une manière qui optimise la définition et la conception globale du système.
- Intégrer la fiabilité, la maintenabilité, la sécurité, la survie, l'ingénierie humaine, et autres facteurs dans l'effort global d'ingénierie en respectant les objectifs en terme de coûts, délais, supportabilité et performances techniques.

Depuis longtemps, des techniques de modélisation ont été utilisées par les ingénieurs système. Parmi ces techniques, on cite SADT et SA/RT, qui datent des années 80, ainsi que de nombreuses approches qui reposent sur les réseaux de Pétri ou les machines à états finis. Ces techniques sont limitées par la difficulté de leur intégration avec d'autres formalismes, comme avec les exigences système.

SADT et SA, initialement utilisées pour l'informatique de gestion, ne couvrent pas en standard les vues dynamiques. En effet, la société Verilog a essayé de faire de la simulation (outil ASA) en ajoutant la possibilité de décrire des diagrammes d'états dans les feuilles de l'arbre

fonctionnel SADT. Cependant, l'implémentation de ces extensions à SADT ou SA était toujours propre à l'éditeur de l'outil associé, donc non standardisé.

1.1.2 Les normes de l'ingénierie systèmes

Une variété de normes d'ingénierie systèmes a été proposée par différents organismes internationaux. La plupart de ces normes ont été évoluées à partir de DoD-MIL-STD 499 [11]. La norme ANSI/EIA 632 a été développée par l'EIA (Electronic Industries Alliance) [12] et adoptée en 1999 par l'ANSI (American National Standards Institute) [13]. Cette norme a été ensuite adaptée par l'IEEE aux systèmes électriques et électroniques sous la forme du standard IEEE 1220 [14]. Le standard ISO 15288 a ensuite été réalisé pour couvrir toutes les phases du cycle de vie du produit, il a été ensuite standardisé par L'IEC (International Electrotechnical Commission) [15] puis par l'IEEE. La figure 1.1 montre les phases d'application de chacun de ces standards.

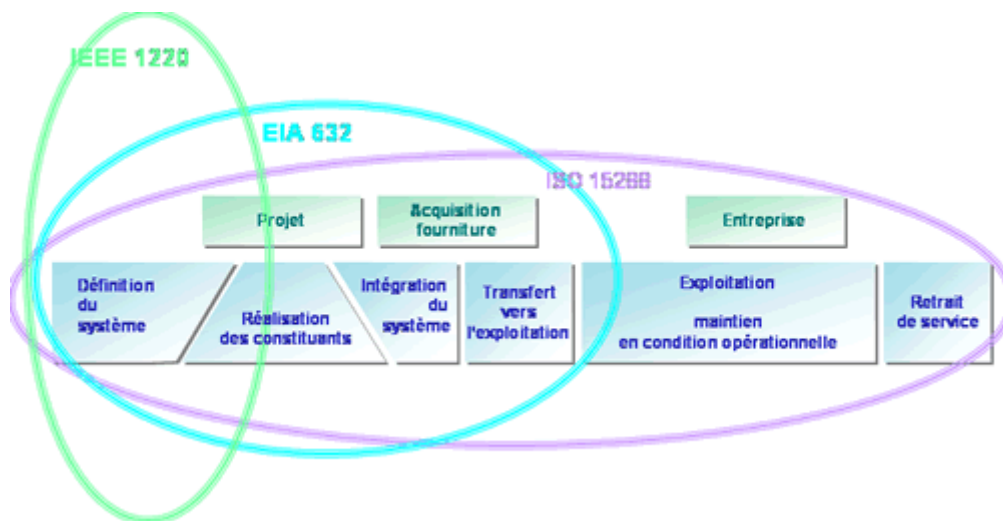


Fig.1.1 : Les activités incluses dans les normes de l'ingénierie systèmes [15]

L'IEEE 1220 est focalisée sur les activités de conception. Tandis que l'EIA 632 étend ses activités à la gestion du projet et à l'acquisition au niveau de l'organisation et va jusqu'aux phases d'intégration et d'exploitation au niveau du système lui-même. Le standard ISO 15288, quant à lui, englobe toutes les activités susceptibles d'exister dans l'entreprise au cours d'un

projet donné y compris les activités de gestion du projet et les activités de conception du système jusqu'à la phase terminale de retrait du système de son environnement d'exploitation.

1.2. L'ingénierie systèmes basée sur les modèles (MBSE)

1.2.1 Généralités

La modélisation est une technique universelle qui permet de comprendre et de simplifier la réalité grâce à l'abstraction. Les principaux objectifs d'un modèle consistent à saisir et énoncer les exigences, ensuite à réfléchir à la conception d'un système, et enfin à maîtriser des systèmes complexes [16]. En outre, la modélisation d'un système est une représentation du résultat souhaité du processus de conception. Comme de nombreux autres aspects de l'ingénierie systèmes, la modélisation est simplement du bon sens. Il est naturel d'imaginer à quoi ressemblera le système avant de s'engager dans sa construction. De nos jours, un modèle de système de haut niveau approprié peut être construit avec un outil informatique qui permet à l'équipe pluridisciplinaire de «visualiser» l'ensemble du système et son environnement [17]. Cela inclut les clients qui peuvent désormais voir plus clairement leur vision. La modélisation permet ainsi de valider la construction du système dans la phase préliminaire de conception. Les auteurs dans [18] ont souligné qu'il est important d'utiliser des modèles pour mieux comprendre les systèmes complexes, faire des expériences, faire fonctionner des systèmes en temps réel et négocier le mode de déploiement du système avec les parties en conflit.

L'ingénierie systèmes basée sur les modèles 'MBSE' est un nouveau paradigme qui facilite la spécification, l'analyse, la conception et la vérification d'un système complexe à l'aide d'un modèle de système intégré avec un outil logiciel dédié [19]. Il joue un rôle important dans la participation des différentes parties prenantes à toutes les phases du développement du produit. La principale contribution de l'MBSE est de faciliter les activités d'ingénierie systèmes (IS) en développant un modèle cohérent unifié en tant qu'artéfact principal [20]. Selon Friedenthal [21],

« Les principaux avantages potentiels de cette approche incluent une communication améliorée entre les parties prenantes et les membres de l'équipe, ainsi qu'une véritable compréhension partagée du domaine, une capture améliorée des connaissances, une précision de la conception, une intégrité sans déconnexion entre les représentations des données, une meilleure traçabilité des informations, une réutilisation améliorée des données, des artefacts et des risques de développement réduits ».

Bahill et Botta [22] ont souligné que l'MBSE repose sur l'application de modèles formels appropriés à un domaine donné. Selon [23], MBSE devrait jouer un rôle croissant dans la pratique de l'ingénierie systèmes et élargir ses domaines de modélisation et d'applications au-delà des systèmes matériels et logiciels, y compris les composants de performance sociale, économique, environnementale et humaine. Les avantages attendus de ce nouveau paradigme incluent l'amélioration de la qualité et de la communication, la traçabilité sans coût, l'augmentation de la productivité.

Cependant, La transition vers MBSE implique un investissement considérable en processus, méthodes, outils et, bien entendu, en formation [24]. L'approche MBSE nécessite une nouvelle façon de penser et un nouvel ensemble de compétences. La communauté travaillant avec les outils de modélisation et les langages doit inclure des experts en langage / outils qui vont développer un modèle de système et seront en mesure de former d'autres membres de l'équipe.

1.2.2 Modèles de développement des cycles de vie en ingénierie système

Un certain nombre de modèles de développement du cycle de vie a été créé et appliqué à des projets de développement des systèmes et de logiciels à grande échelle utilisés par le gouvernement, l'industrie et le milieu universitaire. La plupart de ces modèles se sont basés sur l'un des trois modèles suivants : le modèle de cascade de Royce, le modèle en spirale de Boehm

[25], et le modèle "V" de Forsberg et Moog [26, 27]. Une représentation graphique de chacun de ces modèles de développement du cycle de vie est illustrée dans la figure 1.2 :

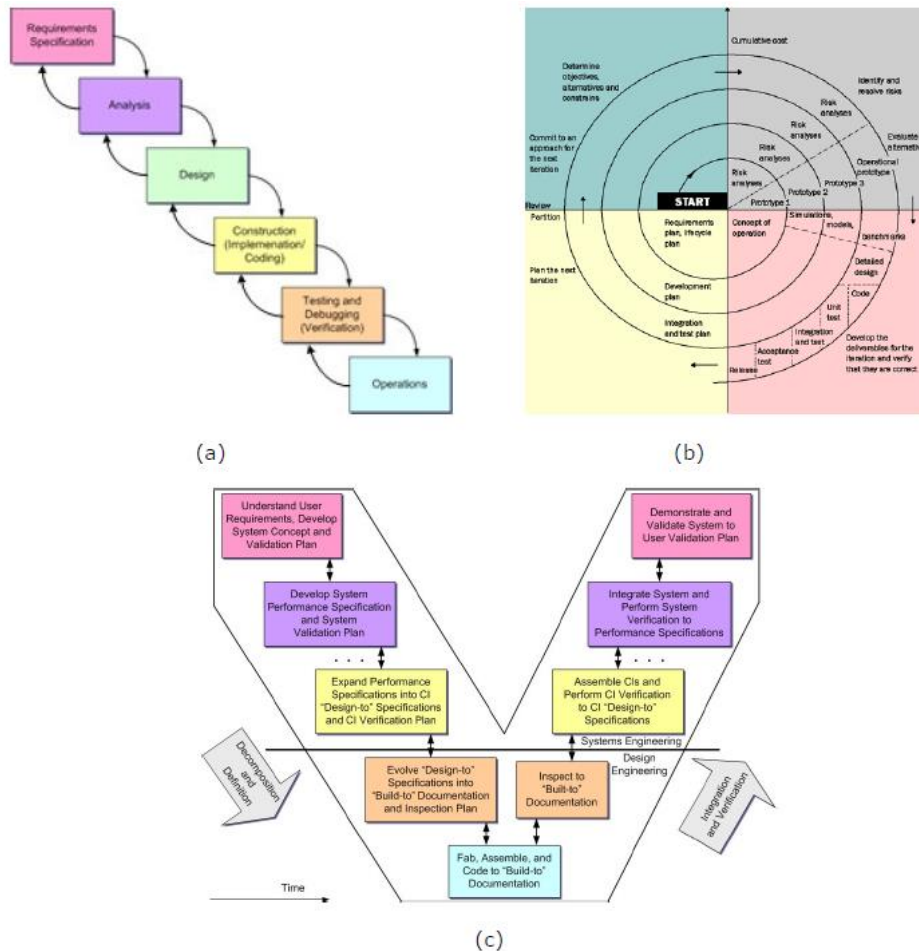


Fig.1.2 : Modèles de développement du cycle de vie : (a) Cascade, (B) Spiral, (c) en V [28]

Les modèles en cascade et en spirale ont été largement utilisés dans les projets de développement de logiciels. Le principe de ces modèles consiste à créer des prototypes et ensuite faire des simulations de ces prototypes itérativement. Dans chaque cycle, une itération est faite pour évaluer des risques et se termine par la résolution de ces risques suivie de l'implémentation et de tests. Le modèle en "V" et les versions modifiées du "V" ont été largement appliqués dans les domaines de l'ingénierie systèmes et le développement de systèmes. Ces modèles peuvent également servir de méta modèle pour le développement du cycle de vie de nouveaux projets.

1.2.3 Présentation du langage SysML

L'OMG (*Object Management Group*) qui s'occupe de la standardisation de certaines technologies objet tel qu'UML (*Unified modeling language*) a proposé de modifier un profil UML afin de couvrir les tâches de l'ingénierie systèmes [29]. UML se définit comme un langage de modélisation graphique et textuel destiné à décrire et à comprendre des besoins, à spécifier et à documenter des systèmes, à concevoir des solutions et à communiquer des points de vue esquisser des architectures logicielles [30].

Pour le passage à la programmation par objets, UML était peu adapté à la modélisation des systèmes complexes dans ses premières versions et donc au support de l'ingénierie systèmes. Les nombreuses tentatives d'utilisation d'UML en IS ont en effet montré que certaines faiblesses d'UML devaient être comblées pour en faire un langage efficace pour les ingénieurs système. On peut citer :

- le besoin de décrire les exigences directement dans le modèle, et d'en assurer la traçabilité vers l'architecture ;
- le besoin de représenter des éléments non-logiciels et d'en spécifier le type (mécanique, circuit, hydraulique, câblage, capteur, etc.) ;
- le besoin de représenter des équations physiques, des contraintes ;
- le besoin de représenter des flux continus (matière, énergie, etc.) ;
- le besoin de représenter des allocations logique/physique, structure/dynamique, etc.

Le travail sur la nouvelle version UML 2.0 a abouti à la définition d'un langage de modélisation très proche du besoin des ingénieurs système, avec les améliorations notables des diagrammes d'UML tels que le diagramme d'activité et le diagramme de séquences, ainsi que la mise au point du diagramme de structure composite. Néanmoins, l'adoption d'UML par les ingénieurs

systèmes restait une barrière à cause de son aspect logiciel non adapté à l'IS [31]. L'OMG a annoncé l'adoption de SysML (system modeling language) en juillet 2006 et la disponibilité de la première version officielle (SysML v. 1.0) en septembre 2007.

SysML est un langage défini comme une extension d'UML 2.0. La partie d'UML 2.0 qui est réutilisée par SysML est appelée UML4SysML (figure 1.3). Les diagrammes UML 2.0 ont été étendus pour subvenir à des besoins spécifiques à l'ingénierie systèmes [32].

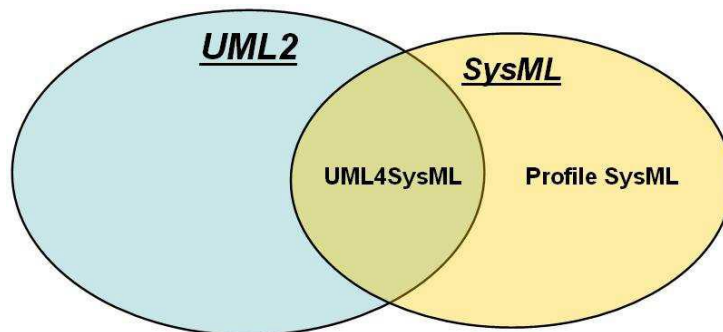


Fig.1.3 : Relation entre SysML et UML 2.0 [32]

SysML prend en charge la spécification, l'analyse, la conception et la vérification des systèmes complexes. Il est considéré comme le langage principal de modélisation pour l'ingénierie systèmes. Selon [33] " malgré le manque de quelques concepts nécessaires de SysML, ce langage graphique a ouvert de nouvelles perspectives pour l'application de l'ingénierie systèmes". Ce langage stocke des données dans un modèle en utilisant une représentation graphique sous forme de diagrammes. Les diagrammes SysML, qui reflètent divers aspects d'un système, sont neufs et sont organisés dans quatre blocs majeurs connus sous le nom « de quatre piliers de SysML » et représentent quatre facettes principales de la modélisation : les exigences du système, la structure, le comportement et les relations paramétriques.

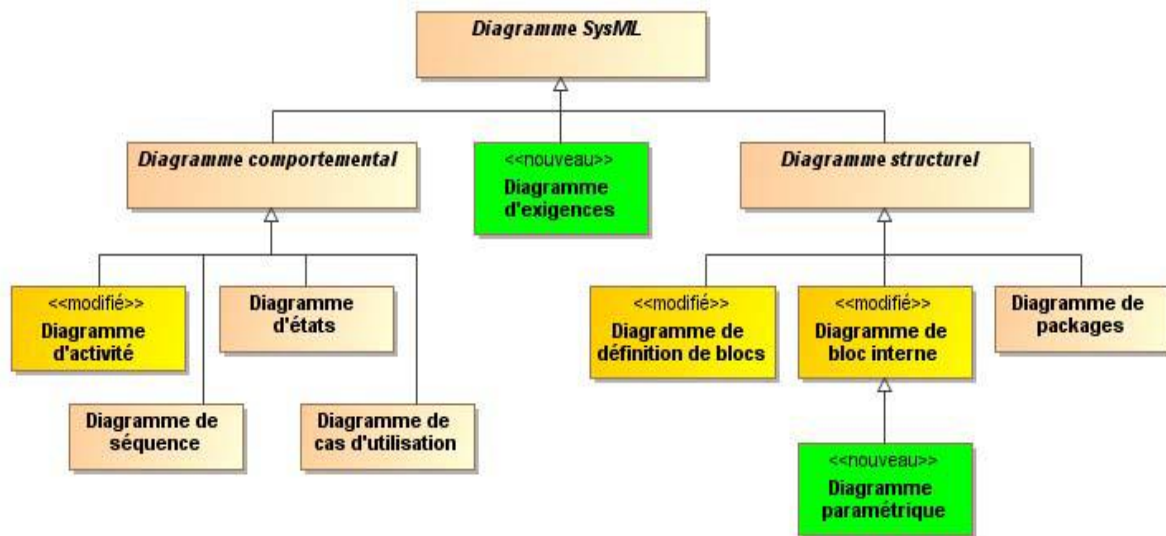


Fig.1.4 : Présentation des neuf diagrammes de SysML [34]

SysML comporte neuf types de diagrammes, chacun d’eux étant dédié à la représentation des concepts particuliers d’un système. Ces types de diagrammes sont répartis par l’OMG en trois grands groupes :

- Les diagrammes comportementaux :

1. diagramme d’activité (Ce diagramme montre l’enchaînement des décisions et des actions au sein d’une activité complexe).
2. diagramme de séquence (Ce diagramme présente l’interaction entre les blocks à travers une séquence verticale des messages passés).
3. diagramme d’états (montre les différents états et transitions possibles des blocs dynamiques)
4. diagramme de cas d’utilisation (ce diagramme montre les interactions fonctionnelles entre les parties externes (acteur) et internes du système).

- un diagramme transverse : le diagramme d’exigences (ce diagramme présente une hiérarchie des exigences du système et leurs relations).

- quatre diagrammes structurels :

1. diagramme de définition de blocs (ce diagramme montre les briques de base statiques : blocs, compositions, associations, attributs, opérations, généralisations, etc.).
2. diagramme de bloc interne (ce diagramme montre l'organisation interne d'un élément statique complexe).
3. diagramme paramétrique (ce diagramme représente les contraintes du système, les équations qui le régissent).
4. diagramme de packages (ce diagramme montre l'organisation logique du modèle et les relations entre packages).

La structure du système est représentée par les diagrammes de blocs. Le diagramme de définition de blocs (*block definition diagram*) décrit la hiérarchie du système et les classifications système/composant. Le diagramme de bloc interne (*internal block diagram*) décrit la structure interne du système en terme de parties, ports et connecteurs. Le diagramme de packages est utilisé pour organiser le modèle.

Les diagrammes comportementaux incluent le diagramme de cas d'utilisation, le diagramme d'activité, le diagramme de séquence et le diagramme de machines à états. Le diagramme de cas d'utilisation fournit une description de haut niveau des fonctionnalités du système. Le diagramme d'activité représente les flots de données et de contrôle entre les actions. Le diagramme de séquence représente les interactions entre les parties du système qui collaborent. Le diagramme de machines d'états décrit les transitions entre états et les actions que le système ou ses parties réalisent en réponse aux événements.

Le diagramme d'exigences capture les hiérarchies d'exigences, ainsi que leurs relations de dérivation, de satisfaction, de vérification et de raffinement. Ces relations fournissent la

capacité de relier les exigences les unes aux autres, ainsi qu'aux éléments de conception et aux cas de tests.

Le diagramme paramétrique permet de représenter des contraintes sur les valeurs de paramètres système telles que la performance, la fiabilité, la masse, etc. Il s'agit d'une spécialisation du diagramme de bloc interne où les seuls blocs utilisables sont des contraintes entre paramètres permettant de représenter graphiquement des équations et relations mathématiques. Ce nouveau diagramme fournit ainsi un support pour les études d'analyse système.

SysML inclut également une relation d'allocation pour représenter différents types d'allocation, comme celles de fonctions à des composants, de composants logiques à composants physiques, ainsi que de software à hardware [34].

Dans nos travaux de recherche, le langage de modélisation SysML est proposé en tant que langage pluridisciplinaire commun pour représenter les dépendances entre les vues multiples du système et pour assurer la traçabilité des données au cours des différentes phases de conception. SysML prend également en charge la conception au niveau des systèmes, ainsi que des mécanismes permettant de personnaliser cet outil pour une analyse de domaines spécifiques. Un autre avantage de SysML est sa conformité aux normes établies dans le secteur industriel, tels que le standard MOF (Meta-Object Facility), le langage de modélisation unifié (UML) et l'interface de message XML (Extensible Markup Language) (XMI), qui permet de soutenir le développement de logiciels et d'importer/exporter des diagrammes paramétriques reliant des données à des outils d'analyse d'ingénierie [35]. Il existe plusieurs outils de modélisation commerciaux de SysML sur le marché tels qu'Artisan Studio, MagicDraw, IBM Rational Rhapsody et Enterprise Architect.

1.2.4 Application de l'MBSE pour le développement des modèles de conception

Un certain nombre de modèles de développement de produits a été créé et appliqué à des projets de développement de systèmes et de logiciels à grande échelle utilisés par les gouvernements, les industries et les universités. La plupart de ces modèles sont basés sur l'un des trois modèles principaux : le modèle en cascade de Royce, la spirale de Boehm et le modèle "V" de Forsberg et Moog. Harmony-SE est un modèle de développement de produit basé sur le modèle en V, utilisé dans une approche de modélisation des demandes de service [36].

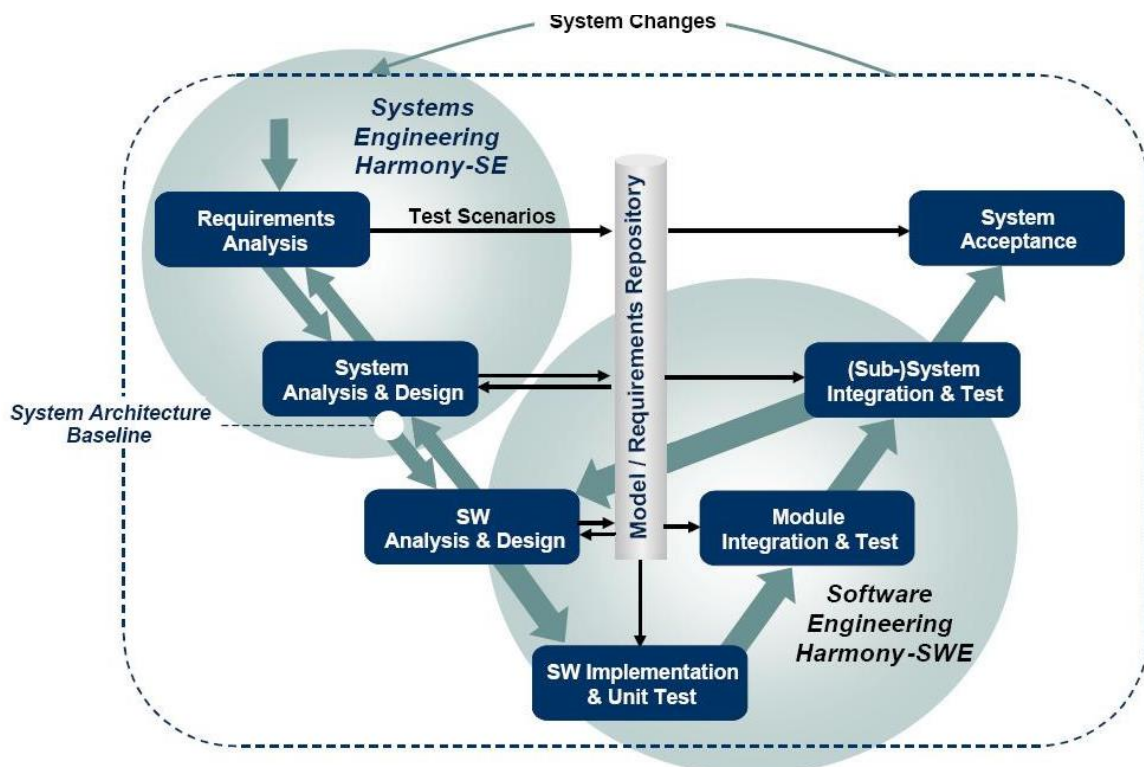


Fig.1.5 : Le modèle Harmony-SE [28]

Dans ce modèle, la structure du système est décrite à l'aide du diagramme de définition de blocs SysML. Les principaux objectifs d'Harmony sont tout d'abord identifier et déduire les fonctionnalités requises du système. Ensuite, identifier les états du système et, enfin, allouer les fonctionnalités du système à une architecture physique.

Un autre modèle de développement de produit appelé RUP SE [37] a été créé pour aider les entreprises à gagner du temps, à réduire les coûts de conception et à améliorer la qualité des systèmes.

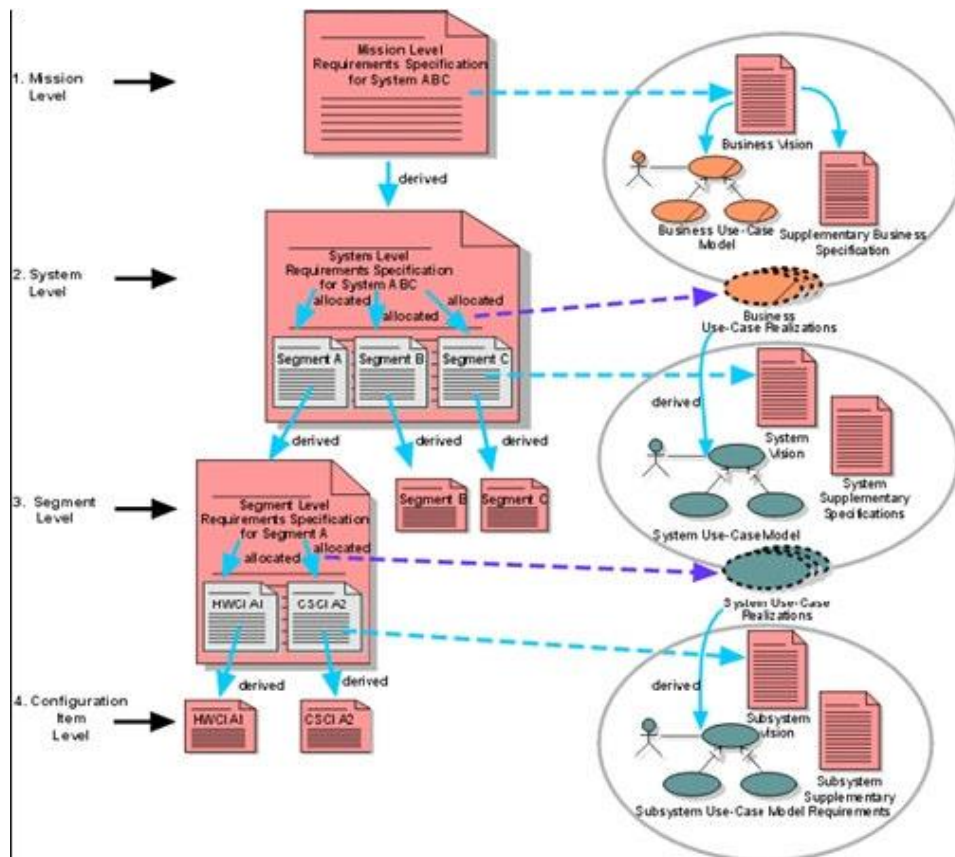


Fig.1.6 : Le modèle RUP SE [28]

RUP SE comprend deux types d'exigences système comme le montre la figure 1.6: les cas d'utilisation, qui capturent les exigences fonctionnelles ; et des exigences supplémentaires, qui couvrent des attributs non fonctionnels telles que la fiabilité et la facilité de maintenance.

Dans le même contexte, Vitech [28] est une méthodologie qui repose sur un processus d'ingénierie systèmes incrémental connu sous le nom du «modèle d'oignon», qui permet d'avoir des solutions intermédiaires complètes pendant le processus de spécification du système. Une représentation visuelle de ce modèle est illustrée dans la figure 1.7 :

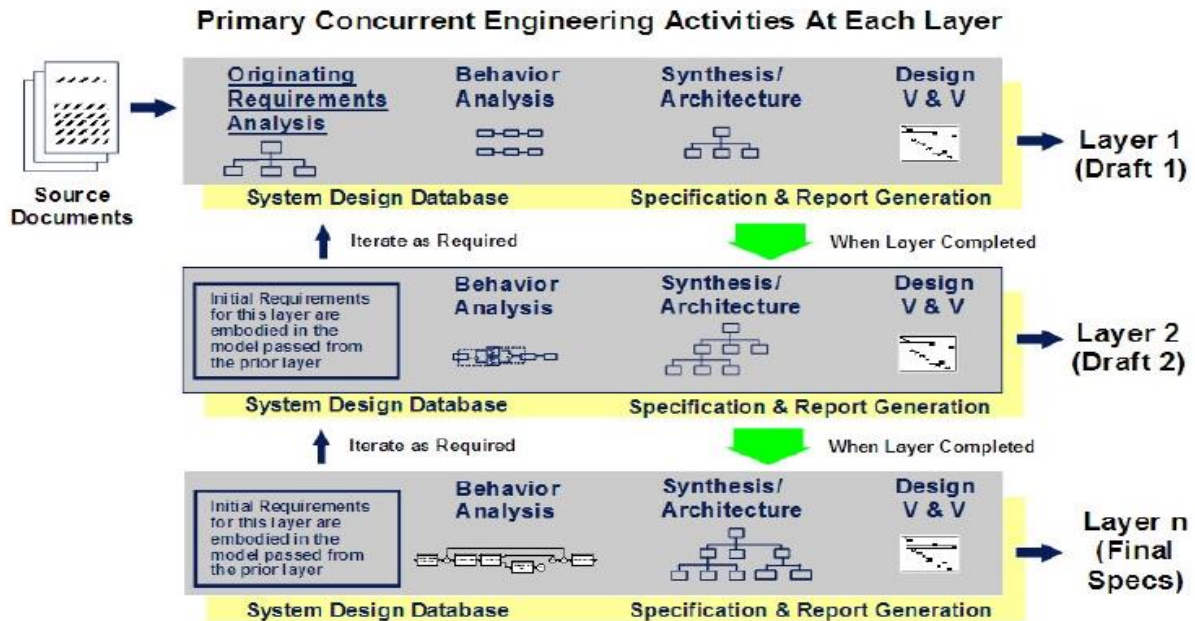


Fig.1.7 : La méthodologie Vitech [28]

Le modèle « oignon » itère les principales activités d'ingénierie systèmes à chaque couche. Selon Childers et Long [38], Lorsqu'une équipe d'ingénieurs système complète avec succès un niveau de conception du système, elle enlève une couche « d'oignon » et commence à explorer la couche suivante. Lorsque l'équipe atteint le niveau de détail souhaité (le centre), sa conception est terminée. Le principal avantage du modèle Onion est d'offrir une approche de conception avec un facteur de risque plus faible.

La méthode d'ingénierie systèmes orientée objet (OOSEM) [39] intègre une approche basée sur les modèles qui utilise SysML pour prendre en charge la spécification, l'analyse, la conception et la vérification des systèmes. OOSEM exploite des méthodes d'ingénierie systèmes "top down" plus traditionnelles et d'autres techniques de modélisation, afin d'aider les architectes à mettre en place des systèmes plus flexibles et extensibles qui peuvent s'adapter à l'évolution de la technologie et aux exigences changeantes. OOSEM est une méthodologie hybride qui s'appuie sur des techniques orientées objet et une fondation d'ingénierie systèmes. Elle présente également des techniques uniques comme indiqué dans la figure 1.8 :

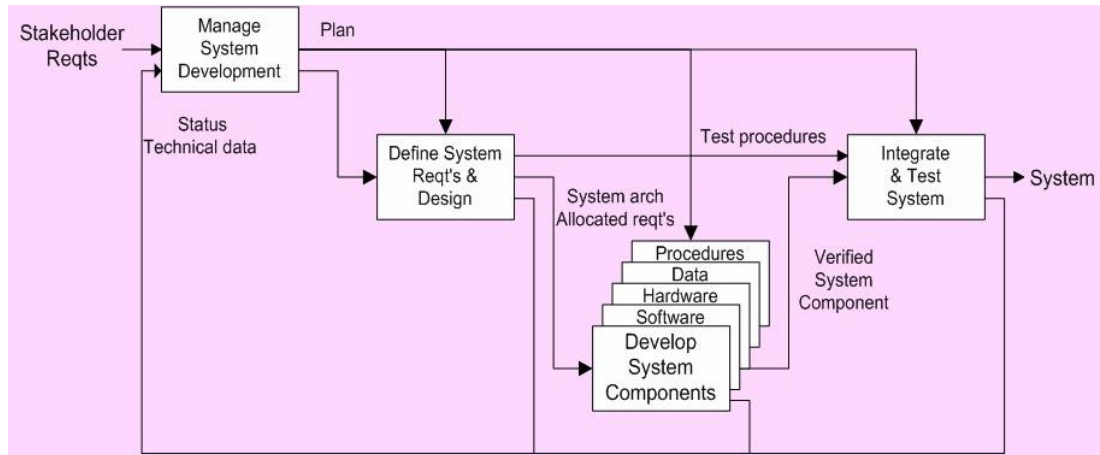


Fig.1.8 : La méthode OOSEM [31]

Cette méthode commence par identifier les exigences des clients, ensuite définir l'architecture fonctionnelle et physique du système et enfin se termine par une phase d'intégration et de test.

Un autre travail met en évidence les avantages de l'utilisation des modèles MBSE dans le centre spatial Johnson de la NASA [40]. En fait, le modèle utilisé dans ce travail intègre des vues des systèmes sous plusieurs perspectives, en capturant les informations de conception de système pour plusieurs parties prenantes.

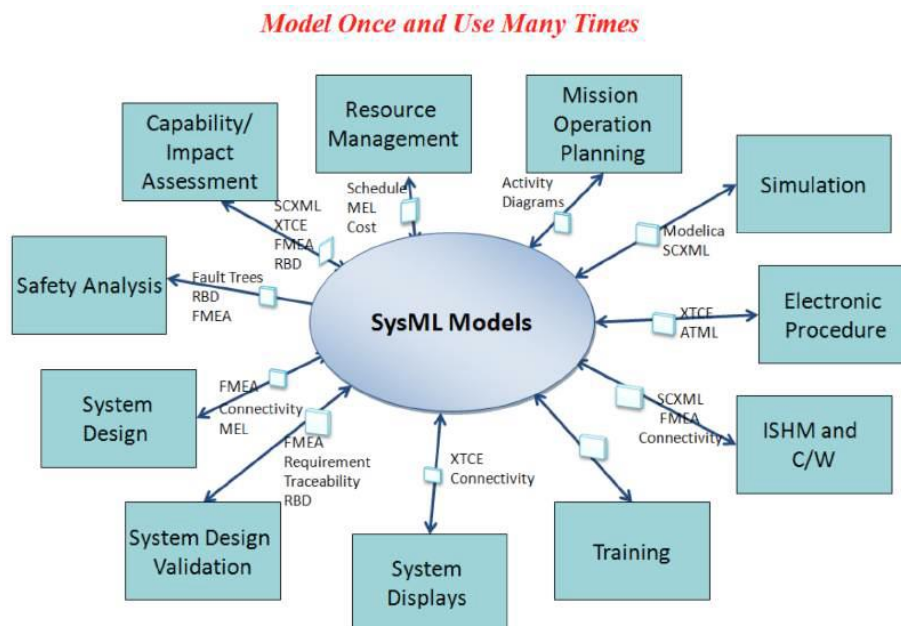


Fig.1.9 : Le modèle de développement des produits de la NASA [40]

Les modèles sont développés à l'aide de diagrammes SysML afin de représenter quatre facettes principales du système : la configuration du système requise, la structure, le comportement et les relations paramétriques. Pour inclure les disciplines en aval dans ce modèle développé, des plug-ins ont été créés pour permettre à plusieurs parties prenantes du système d'extraire des attributs pour leur propre utilisation.

Les recherches menées dans [41] ont mis au point une méthodologie de développement de produits pour la conception de systèmes mécatroniques. Cette méthodologie est composée d'une phase de boîte noire, qui est un point de vue externe du système tandis que la boîte blanche est réalisée avec un point de vue interne.

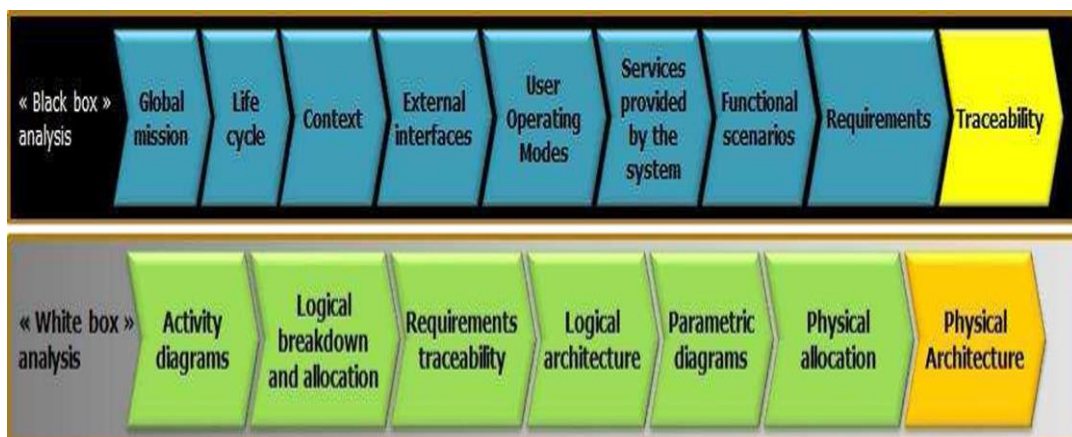


Fig.1.10 : La méthodologie de conception des systèmes mécatroniques [41]

La plupart des travaux de recherche basés sur les modèles et les méthodologies ci-dessus pour la modélisation de systèmes complexes au stade de la conception initiale ont été effectués en utilisant les approches traditionnelles nommées (Point-Based engineering). En effet, les auteurs dans [42] ont travaillé sur la réduction du temps de calcul et l'amélioration de la recherche des configurations optimales des systèmes mécatroniques. À cet effet, un nouvel indicateur multicritères d'évaluation des performances du système, appelé indicateur de conception mécatronique (MDI), est utilisé pour aider les concepteurs de systèmes mécatroniques à localiser de manière interactive les points de départ de l'optimisation en phase d'exploration de

conception. En outre, la méthode MBSE a été utilisée pour modéliser et optimiser les performances des véhicules électriques (VE) au niveau de la conception. Les structures comportementales et fonctionnelles sont modélisées à l'aide de diagrammes SysML. Le choix de la meilleure architecture du modèle est effectué à l'aide du langage Modelica et de modèles de substitution. Une autre approche pour optimiser les véhicules électriques est basée sur la méthode de partitionnement à base de contraintes (CBPM) [43] pour décomposer le problème de conception mécatronique en sous-problèmes et pour optimiser le système global. Pour cela, des modèles d'analyse ont été développés à l'aide de Modelica pour vérifier les performances globales du système. Les partitions sont ensuite implémentées dans le cadre MDO (Multidisciplinary Design Optimization) pour trouver les solutions optimales. L'approche proposée aide les concepteurs de systèmes mécatroniques à vérifier et à valider la conception de ces systèmes et donc à prendre des décisions de manière efficace. Les auteurs dans [44] ont présenté un exemple d'une application de la méthodologie OOSEM pour concevoir un engin spatial. L'étude révèle des bénéfices potentiels encourageants de l'utilisation de cette méthodologie MBSE, comme l'amélioration de la communication entre les concepteurs de modèles et les parties prenantes, la représentation cohérente et complète des modèles de système dans différentes missions et phases, la réduction des erreurs et de l'ambiguïté, la réduction des coûts de maintenance, du temps de conception et des ressources allouées. Simpkins et al. [45] ont présenté une application pratique de MBSE, en utilisant la méthodologie Vitech, conduisant ainsi à une solution intégrée et convergente pour un système de stationnement automatisé. Les principaux avantages signalés impliquent une meilleure compréhension du problème, une réponse plus rapide aux enquêtes des parties prenantes, une traçabilité plus rigoureuse et une vérification automatique de la documentation. Soyler [46] a

proposé l'adoption de l'approche holistique MBSE pour capter la structure et le comportement des systèmes de gestion des catastrophes naturelles et pour faire face à leur complexité.

Malgré le grand nombre des travaux réalisés sur le développement des systèmes mécatroniques, peu d'études ont envisagé l'évaluation intégrée de l'aspect qualité, fiabilité et fabricabilité pour l'étude de la robustesse des produits au stade de la conception. En intégrant une approche qui se base sur l'ingénierie systèmes basée sur les modèles et une approche concourante qui traite plusieurs solutions simultanément, les ingénieurs seront en mesure d'intégrer plusieurs aspects pour améliorer les performances du système et réduire les délais et les coûts de production dès les premières étapes de la conception. La méthodologie combinée d'ingénierie simultanée et d'ingénierie systèmes basée sur des ensembles offre un grand potentiel pour gérer les aspects de fabricabilité et de fiabilité, en raison de la capacité à développer des connaissances en parallèle, en particulier pour les technologies avancées et non mures [47]. Ceci constitue l'intérêt de notre travail de recherche.

Les différents aspects de MBSE qui ont été discutés dans les sections précédentes (les méthodes, les outils de modélisation, les applications, etc.) peuvent être considérés comme des dimensions complémentaires qui doivent fonctionner ensemble pour aboutir à un système réussi. Ce succès est mesuré par la satisfaction des attentes des parties prenantes. L'avenir de MBSE sera facilité par les technologies de l'information en évolution continue (informatique, capacités de stockage et d'analyse, capacités distribuées, réseaux virtuels, etc.) ainsi que par le profil d'ingénieurs systèmes modernes.

1.3. Utilisation de l'approche Lean pour le développement des produits complexes

1.3.1 Généralités at applications

La capacité de développement de produits complexes tels que les systèmes mécatroniques avec des délais courts devient essentielle pour les entreprises, car la concurrence mondiale croissante

et la segmentation des marchés accélèrent le rythme des changements dans de nombreux secteurs [48]. Cependant, les entreprises qui ont structuré le processus de développement de leurs produits (PDP) basé sur des modèles traditionnels peuvent être désavantagées en ce qui concerne les dimensions importantes telles que l'agilité, la flexibilité et la productivité. Parmi ces modèles, on trouve l'approche par point (citée dans les sections précédentes) qui consiste à explorer quelques concepts ensuite sélectionner une seule solution de l'espace de conception puis faire des modifications après chaque itération comme le montre la figure 1.11.

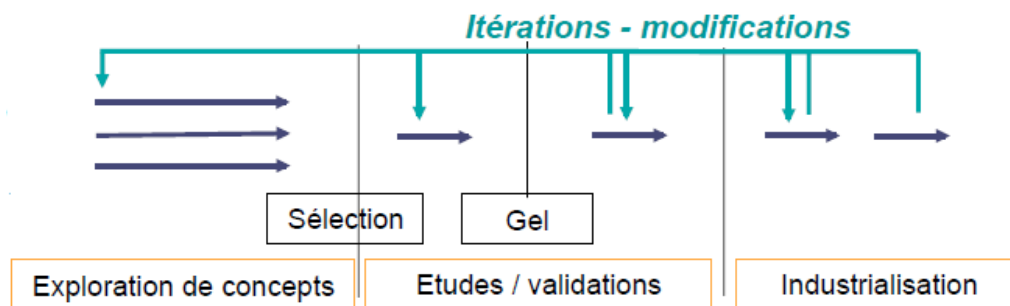


Fig.1.11: L'approche par point (Point-Based Engineering) [49]

Des problèmes avec une telle approche surviennent lorsque des ingénieurs tentent de travailler simultanément avec d'autres membres de l'équipe de développement. Au fur et à mesure que la conception passe d'un groupe à l'autre pour une critique de différents points de vue fonctionnels, chaque changement entraîne des modifications et des analyses supplémentaires. En outre, les modèles de PDP traditionnels reposent sur l'ingénierie séquentielle entraînant un certain nombre de problèmes couramment rencontrés dans les entreprises; certains d'entre eux sont la surcharge du travail des concepteurs et des ingénieurs qui effectuent fréquemment des tâches inutiles, un modèle PDP qui n'est pas clairement défini et bien compris par les concepteurs et des dépassements de coûts du projet [49].

Pour remédier à ces lacunes, l'application des principes de la pensée « Lean Product Development » dans les industries automobiles leur a permis d'être leader dans le processus

de d veloppement des produits. Browning [50] souligne que le concept de Lean Product Development (LPD) d passe l'objectif d' limination des d chets: il vise   maximiser la valeur ajout e pour les clients, les actionnaires, les employ s, la soci t  et fournisseurs. La pens e Lean repose sur plusieurs m thodes et outils. L'ing nierie concourante est l'un des piliers de la pens e Lean qui consiste   engager simultan ment tous les acteurs d s le d but d'un projet pour la compr hension des objectifs recherch s et de l'ensemble des activit s qui devront  tre r alis es. La compr hension globale qui en r sulte facilite la d tection pr coce des probl mes potentiels, permet de mettre en  vidences des interd pendances complexes ou floues et aboutit   un accroissement de la motivation des ressources humaines. Selon Abdalla [51] «L'ing nierie concurrente (CE) est la prise en compte des facteurs associ s au cycle de vie du produit lors de la phase de conception. Ces facteurs incluent la fonctionnalit  du produit, la fabrication, l'assemblage, les tests, la maintenance, la fiabilit , le co t et la qualit . »

En ing nierie int gr e ou simultan e, les diff rentes t ches se recouvrent ce qui se traduit par un gain de temps pouvant  tre d di    d'autres activit s du processus de conception et de d veloppement de produits.

La conception bas e sur les ensembles (Set-Based Design) est un autre pilier de la pens e Lean. En tant que processus de synth se  volutif, l'avantage principal de SBD est de fournir des possibilit s continues de communiquer simultan ment les diff rentes alternatives au cours du processus. Pour ce faire, il est possible de donner   chaque participant la possibilit  de prendre une d cision au d but de la phase de conception, puis de la r duire progressivement pour aboutir   la meilleure solution [52].

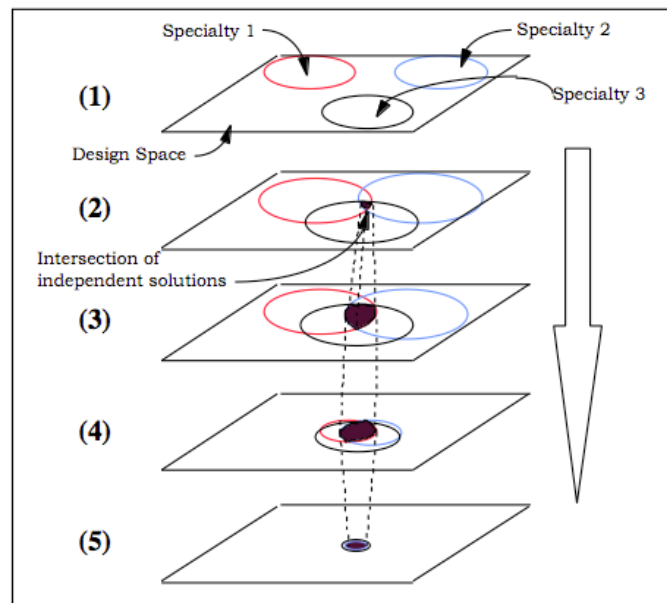


Fig.1.12: L'approche Set-Based Design [52]

Les participants interagissent les uns avec les autres pour ajouter ou supprimer des composants dans l'espace de solution afin de pouvoir développer un ensemble satisfaisant de combinaisons de conception possibles tout en minimisant les chances d'éliminer une alternative réalisable. Par conséquent, SBD contribue à améliorer l'efficacité et l'efficacité d'un processus de développement de produit, notamment en terme de qualité du produit final et de délai d'exécution [53,54].

1.3.2 L'approche SBCE

En se basant sur la pensée Lean, Sobek et al. [55] ont proposé le cadre d'ingénierie simultanée basée sur les ensembles (SBCE) à partir d'observations de Toyota, qui fournit une liste de principes pratiques de l'approche basée sur les ensembles dans un processus de développement de produit. En effet, Toyota répond aux besoins du marché et développe des produits plus rapidement que les autres constructeurs automobiles. La figure 1.13 illustre cette approche technique simultanée.

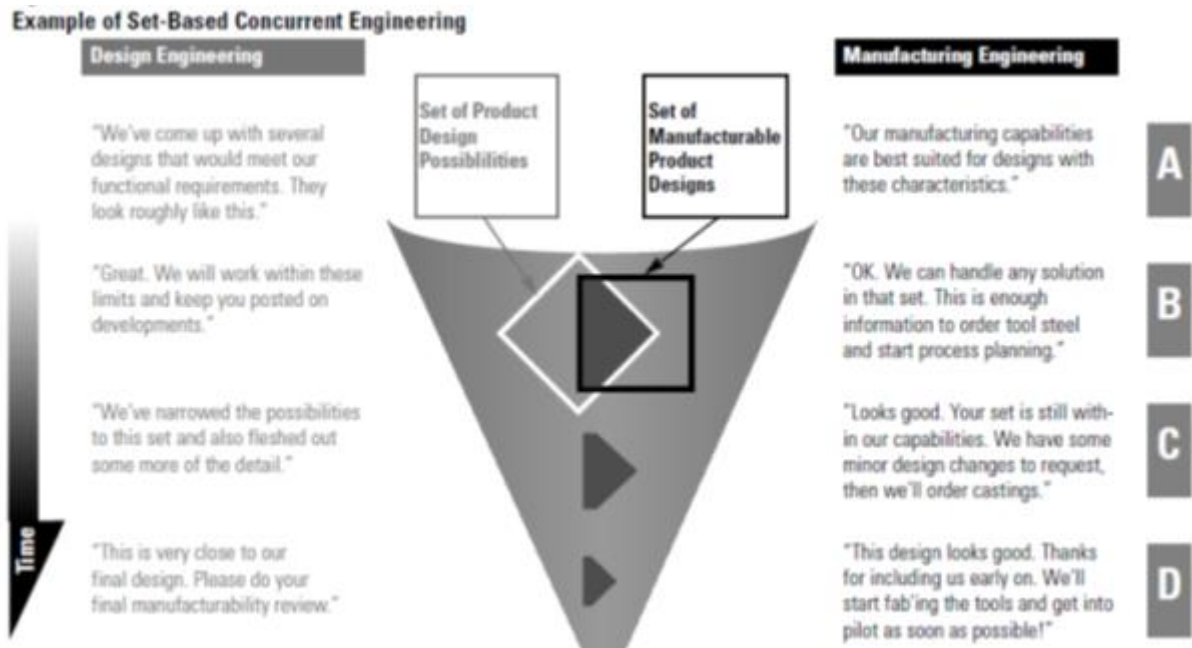


Fig.1.13 : Exemple d'application de l'approche SBCE dans la société Toyota [55]

L'approche SBCE développée est une approche efficace qui a contribué à la croissance de cette société et qui a influencé le processus de développement des produits dans d'autres entreprises. Une étude menée par huit entreprises automobiles en Afrique du Sud sur le système de développement de produits Lean (LPDS) de Toyota montre que certains aspects de l'ingénierie des produits ont été influencés par la pensée Lean dans ces entreprises [56].

SBCE pourrait être défini comme le processus par lequel les participants à la conception pratiquent cette approche en raisonnant, développant et communiquant des ensembles de solutions en parallèle. Au fur et à mesure de la conception, ils restreignent progressivement leurs ensembles de solutions respectives en fonction des connaissances acquises. Au fur et à mesure qu'ils éliminent des solutions, ils s'engagent à rester dans les limites établies afin que les autres puissent compter sur leur communication. Les principaux principes du SBCE sont illustrés dans la figure 1.14 [57] :

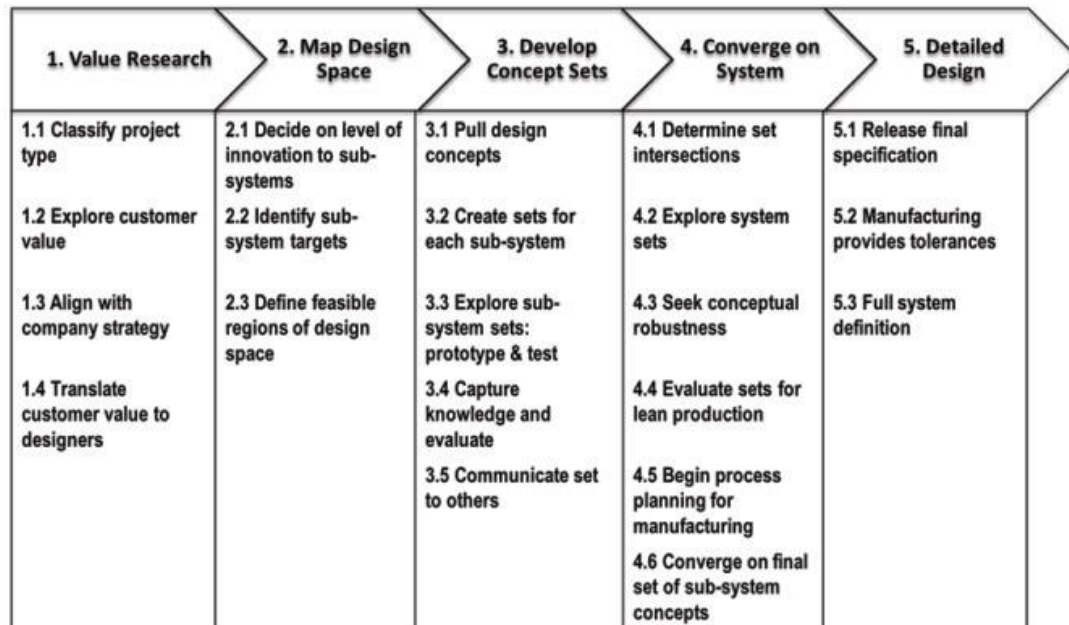


Fig.1.14 : les étapes du SBCE [57]

1. Recherche de valeur. Le projet est classé et défini en fonction du niveau d'innovation incorporé. Les valeurs du client seront également identifiées afin de les évaluer.

2. cartographier l'espace de conception. Les participants à la conception ou les équipes de sous-systèmes définissent la portée des travaux de conception requise, ainsi que les options / régions de conception réalisables. Cela comprend le niveau d'innovation du système et des sous-systèmes.

3. Développement d'un ensemble de concepts. Chaque participant ou équipe de sous-systèmes développe et teste un ensemble de solutions conceptuelles possibles pour la conception de sous-systèmes. A ce stade, les travaux incluent l'exploration de l'ensemble des sous-systèmes. L'exploration est faite soit par la simulation, soit par le prototypage ou par le test. Les connaissances créées au cours de ces activités sont capturées et utilisées pour évaluer différents ensembles de solutions. Ces ensembles de solutions sont communiqués à d'autres équipes afin de recevoir des informations en retour pour analyser et comprendre les contraintes.

4. Convergence des concepts. L'intégration des sous-systèmes est explorée et les systèmes intégrés sont testés. Sur la base des connaissances acquises au cours de cette phase, les solutions système les plus faibles seront filtrées, ce qui permettra à une solution de conception de produit optimale d'entrer dans la phase finale. L'élimination a lieu à la lumière de plusieurs activités, notamment l'évaluation de la robustesse, l'évaluation des coûts et la convergence progressive vers une solution.

5. Conception détaillée. Le dernier ensemble est identifié et les spécifications détaillées sont communiquées aux ingénieurs métiers.

1.3.3 Application du SBCE

Un nombre limité d'études de cas de l'approche SBCE a été réalisé afin d'identifier le potentiel et les avantages de cette approche pour les industries [58]. En effet les auteurs dans [59] ont développé une approche basée sur les flux d'échanges pour optimiser le choix de l'architecture système. Tout d'abord, pour automatiser le processus de simulation et pour générer l'ensemble des solutions systèmes, l'outil Modelica a été connecté à un outil de conception multidisciplinaire (MDO). Pour la phase d'intégration, les principes de l'SBCE sont utilisés pour diminuer l'espace de conception en intégrant une analyse de domaine spécifique, telles que la performance et la fiabilité, afin de converger vers un ensemble de solutions optimales.

Un autre modèle d'évaluation de l'approche SBCE a été conduit dans [60]. Ce modèle montre que le développement de plusieurs alternatives de conception en parallèle pourrait générer une valeur significative pour l'entreprise en intégrant la modularité dans l'architecture du produit afin de basculer entre les alternatives de conception.

Dans la même perspective, l'application de l'approche SBCE dans [61] illustre une étude de cas d'une pompe à jet utilisant les principes de l'SBCE.

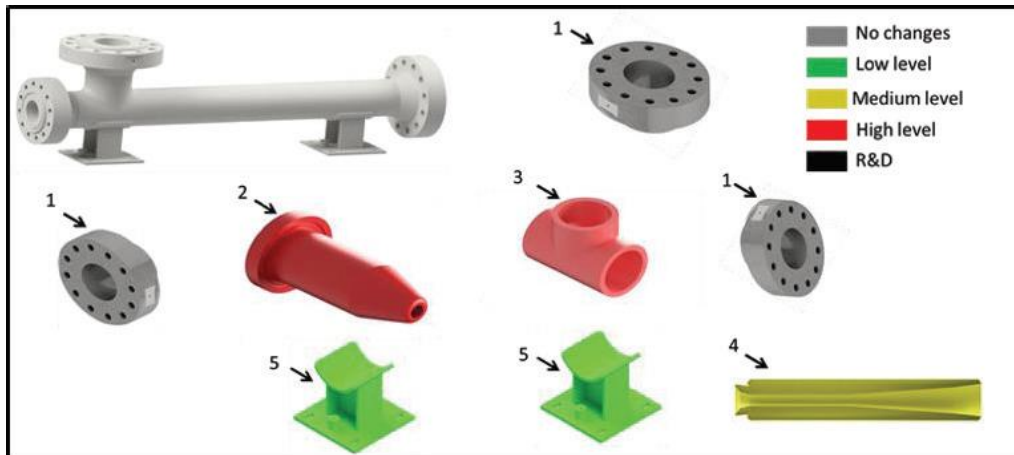


Fig.1.15 : Présentation d'un cas d'étude de la pompe à jet [61]

Cette étude de cas a été bénéfique pour la société en améliorant son processus de développement et en offrant un espace pour explorer des conceptions alternatives sous différents angles tels que la performance, la fabricabilité et le coût du produit. L'étude de cas a identifié plusieurs avantages concrets comme l'amélioration de l'innovation de produit, l'augmentation de la performance et la diminution du coût de fabrication.

Dans le même contexte, Kerga et al. [62] ont développé un jeu autour des principes de l'approche SBCE afin de mieux comprendre et de mieux percevoir ses avantages. Ce jeu compare l'approche traditionnelle par point et l'approche SBCE à travers des mesures de performance. Le résultat de ce jeu montre que l'approche SBCE est plus efficace et pourrait converger vers la solution optimale avec un coût et un délai minimal.

Les auteurs dans [63] ont étudié l'intégration des principes de l'approche SBCE dans le modèle de développement des produits de l'industrie Rolls-Royce. Cette intégration est illustrée dans la figure 1.16. En effet, l'objectif de l'équipe de recherche de cette société est la transformation du processus de développement vers un processus qui se base sur la pensée Lean. Ceci va engendrer une amélioration du processus de développement du produit en terme de niveau d'innovation et de réduction du risque de reconception.

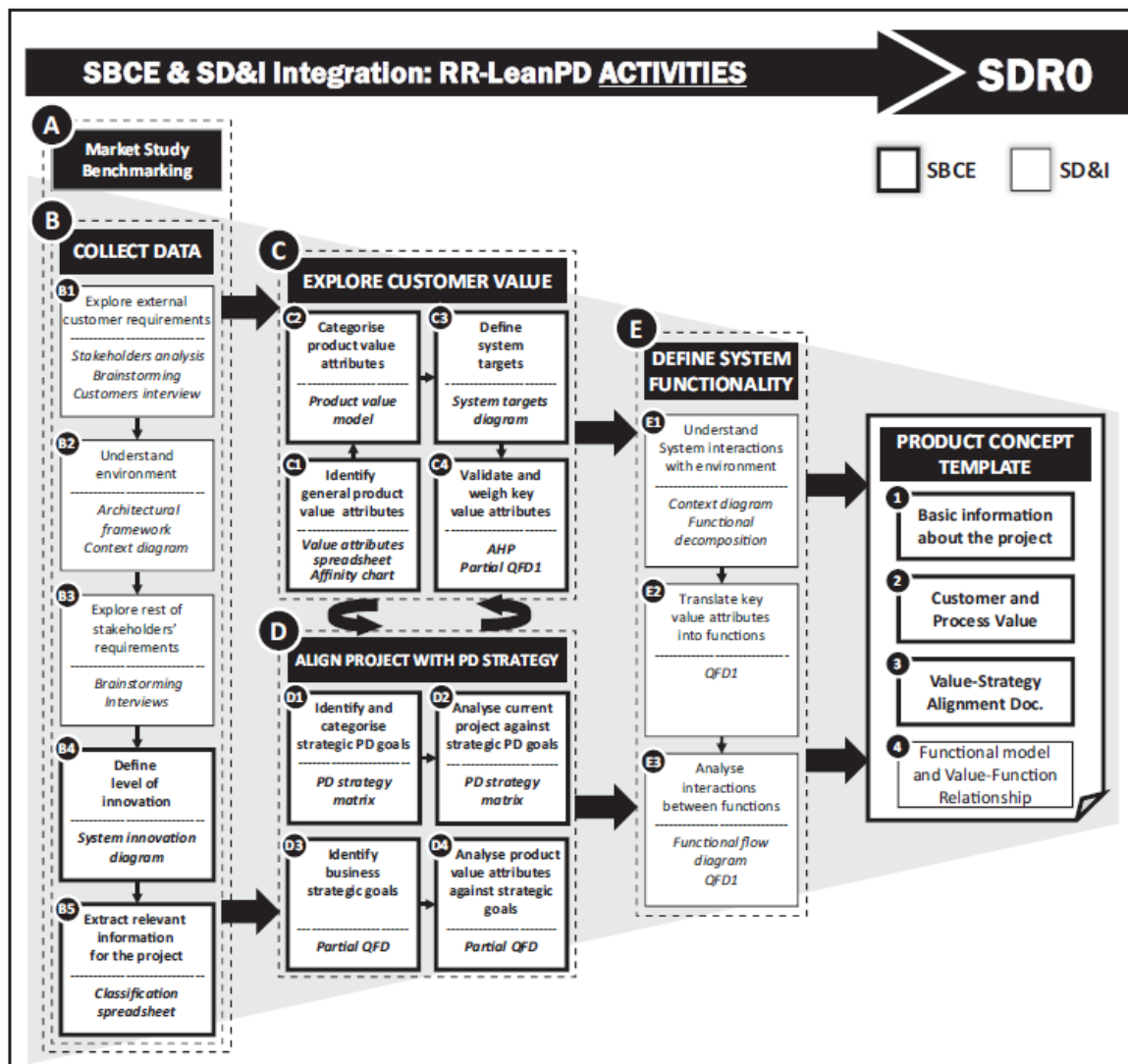


Fig.1.16 : Application du SBCE dans l'entreprise Rolls-Royce [63]

Afin de surmonter les incertitudes au début de la phase de conception, les chercheurs dans [64] ont développé un système informatique appelé 'conception basée sur les ensembles de préférences' qui permet de mettre en œuvre les principes de l'approche SBCE. Ce système est basé sur des fonctions permettant de représenter un ensemble discret de solutions et de respecter la limite des compromis du client. L'avantage de ce système est de filtrer automatiquement les solutions faibles ou coûteuses de l'espace de conception.

Dans les travaux de recherche réalisés ci-dessus, il n'y a pas de détails sur une application systématique des principes de l'approche SBCE identifiant ces avantages tangibles pour les

industries. En outre, ces travaux de recherche ne prennent pas en compte plusieurs aspects du syst me (fiabilit  de la fabricabilit , performances, etc.) en m me temps lors de la phase de d veloppement. L' tude est limit e   un seul domaine d'analyse. Une autre limite est l'absence de m thode claire pour extraire les donn es n cessaires de l'environnement de d veloppement afin de r aliser une analyse de domaine sp cifique. Enfin, pour le quatri me principe de l'approche SBCE, "convergence sur le syst me", les travaux de recherche men s dans les  tudes ci-dessus ne prennent pas en compte une m thode algorithmique permettant de converger vers un ensemble r duit de solutions optimales. Cela augmentera le temps de d veloppement. Pour clarifier les lacunes des travaux de recherche men s, les objectifs de ce travail sont pr sent s   travers les questions ci-dessous:

- Comment d velopper une m thode de conception de produit bas e sur les principes SBCE qui automatise la cartographie de l'espace de conception?
- Comment effectuer une analyse de domaine sp cifique en terme de fabricabilit  et de fiabilit  en se basant sur les donn es extraites des diagrammes SysML?

CHAPITRE 2.

Etude comparative sur les méthodes intégrant la fabricabilité

2.1.	Aperçu sur la fabricabilité.....	38
2.2.	Les critères d'évaluation de la fabricabilité :.....	39
2.3.	Application de ces critères dans les travaux basés sur la fabricabilité	40
2.3.1	Reconnaissance des entités géométriques	40
2.3.2	La visualisation et le retour d'information (feedback).....	43
2.3.3	Les directives de la conception pour la fabrication (DFM).....	48
2.4.	Tableau comparatif sur les méthodes intégrant la fabricabilité	53
2.5.	Contribution de notre travail pour l'étude de la fabricabilité.....	55

Ce chapitre est dédié à l'étude de la fabricabilité dès la phase amont de la conception. La section 2.1 définit la fabricabilité. La section 2.2 présente les critères d'évaluation de la fabricabilité. La section 2.3 traitent quelques travaux réalisés avec ses critères. La section 2.4 est un tableau récapitulatif contenant les avantages et les inconvénients de chaque méthode intégrant la fabricabilité et la section 2.5 présente l'originalité de notre travail sur la fabricabilité.

2.1. Aperçu sur la fabricabilité

La fabricabilité est l'aptitude d'un produit à être réalisé facilement, de façon constante et avec une qualité élevée [65]. La notion de fabricabilité comprend la prise en compte d'un ensemble de contraintes liées à la production d'un produit qui inclut les processus de fabrication et d'assemblage. En effet, la conception basée sur la fabricabilité influe sur le développement des produits afin d'avoir de résultats satisfaisants en fabrication.

La notion de fabricabilité consiste à :

- Intervenir dès la phase préliminaire du cycle de développement des produits et intégrer des procédés rapides à implémenter dans le but de répondre aux appels d'offres avec des délais courts ;
- Bien maîtriser les coûts qui proviennent des procédés de fabrication. Ceci revient à utiliser des procédés de fabrication moins coûteux et ayant un bon rapport qualité/coût.
- Bien maîtriser la qualité dès la phase de conception. Ceci implique une adéquation entre la qualité requise pour le bon fonctionnement de la pièce finie et celle obtenue par les procédés de fabrication.

Gupta et al [66] définissent la notion de la fabricabilité comme la considération simultanée des objectifs de la conception et des contraintes de la fabrication dans le but d'identifier les

problèmes liés à la fabrication et de faciliter leur résolution lors de la conception. Ceci incite le concepteur à produire une conception qui tient compte des contraintes de formes, de tolérance et de l'état de surface. L'évaluation de la fabricabilité revient à quantifier des indicateurs ou des critères dits de fabricabilité, permettant par exemple de comparer différentes solutions.

2.2. Les critères d'évaluation de la fabricabilité :

Les travaux de recherche menés sur la fabricabilité se focalisent sur des critères d'évaluation afin de caractériser le fait que la pièce est facilement fabricable ou non. Ceci implique le coût, le temps de fabrication et la technologie utilisée. Parmi ces critères, nous en citons les suivants :

- Le critère de reconnaissance des entités de la pièce : Ce critère est utilisé dans plusieurs travaux de recherche. En effet, l'interprétation de la forme des entités est un critère important pour analyser la conception donc la fabricabilité de la pièce et intégrer le lien entre la conception et la fabrication. En effet, l'entité est définie comme suit : « un groupement sémantique caractérisé par un ensemble de paramètres, utilisé pour décrire un objet indécomposable dans le raisonnement relatif à une ou plusieurs activités liées à la conception et l'utilisation des produits et des systèmes de production ».
- Le critère de visualisation et de retour d'information (feedback) : Ce critère consiste à faire une analyse de la conception puis retourner cette analyse au concepteur sous forme de texte ou d'image afin de détecter plus facilement le problème lié à la fabricabilité de manière instantanée.
- Le critère des directives de la conception pour la fabrication (DFM). En effet, (DFM) était une tendance dès la seconde guerre mondiale [67]. Ces directives représentent de nombreux efforts pour rassembler les connaissances et les expériences d'experts en sites de fabrication. Un ensemble de directives pour un processus de fabrication contient les connaissances nécessaires qui devraient être satisfaites pour fabriquer une pièce par le processus correspondant [68]; les

directives comprennent des recommandations géométriques et des suggestions de matière première, ainsi qu'une gamme complète de spécifications de tolérance pour une conception.

2.3. Application de ces critères dans les travaux basés sur la fabricabilité

2.3.1 Reconnaissance des entités géométriques

La reconnaissance des entités géométriques permet de traduire les caractéristiques de l'entité (dimensions, tolérances géométriques et dimensionnelles, ...) en informations de fabrication. Avec l'utilisation généralisée des systèmes de CAO, des études intégrant des outils d'évaluation de fabricabilité autonomes ont également vu le jour grâce aux méthodologies de reconnaissance des entités. Medani et Ratchev [69] ont étudié une méthode d'extraction d'entités de conception afin de mettre en œuvre une évaluation rapide de la fabricabilité sur le Web. Dans cette recherche, un système d'agents a été proposé pour faciliter la prise de décision de la fabricabilité dès les premières étapes de la conception. Ce système est basé sur des taxonomies de modèles représentant des informations sur les produits, les pièces, les processus et les ressources de fabrication. L'objectif de ce système développé est de faire correspondre les exigences de conception aux capacités des ressources de fabrication dans un environnement de l'entreprise étendu. Cependant, le travail proposé n'analyse que les processus d'usinage et ne comporte ni liaison ni extension à d'autres processus.

Han et al. [70] ont développé une méthodologie d'optimisation des coûts de fabrication en utilisant la reconnaissance intégrée des entités de conceptions et des gammes d'usinage. Pour la reconnaissance des entités, ils utilisent un outil appelé « l'Integrated Feature Finder (IF²) ». À l'aide des protocoles STEP AP203 et AP 224, ils reconnaissent avec succès les entités d'usinage et les interprètent pour définir de façon optimale le plan de configuration et la séquence d'usinage au moyen d'une stratégie de recherche des entités à deux couches permettant d'optimiser les coûts.

Pham et Ji [71] ont développé un système expert qui permet d'analyser la fabricabilité d'une pièce. Le système proposé extrait les entités géométriques d'une pièce à l'aide du modèle CAO Pro / Engineer et génère ensuite des informations sur les séquences d'usinage possibles, les types d'outils de coupe et les paramètres de coupe. Pour générer ces informations, les chercheurs utilisent une approche basée sur les entités pour la récupération de données de surfaces, d'arêtes et de sommets. En utilisant les données dimensionnelles extraites, ils génèrent les sélections d'outils correspondantes et effectuent des calculs de temps d'usinage des entités de la pièce. Cependant, le travail est limité au processus d'usinage et ne concerne pas les géométries arbitraires. Ils supposent que tous les paramètres sont complètement prédéfinis. En outre, le cadre proposé ne vise que les conceptions de pièces présumées fabricables.

Au-delà des processus d'usinage, Chen et al. [72] développent une approche d'extraction des entités géométriques liées au processus de moulage par injection. En effet, ce procédé de fabrication convient aux modèles à géométrie arbitraire. Pour évaluer la fabricabilité, un raisonnement géométrique sur les interactions d'entités est traité dans l'approche proposée. Ce raisonnement définit les relations spatiales entre les entités afin de déterminer la capacité de moulage d'une forme complexe en terme d'entités individuelles et de l'ensemble de la pièce. Cependant, ce raisonnement ne fournit pas de calcul tangible pour analyser la fabricabilité basée sur l'algorithme de relation spatiale des entités.

Lockett et Guenov [73] développent une approche de reconnaissance des entités de moulage par injection afin d'aider les concepteurs à prendre en compte les contraintes de fabrication dès les premières phases de conception. Pour les pièces moulées et coulées à paroi mince produites à l'aide du protocole STEP AP203, ils proposent une approche basée sur les surfaces moyennes qui reconnaît graphiquement les lignes et les sommets et détermine leur adjacence, comme illustré dans la figure 2.1.

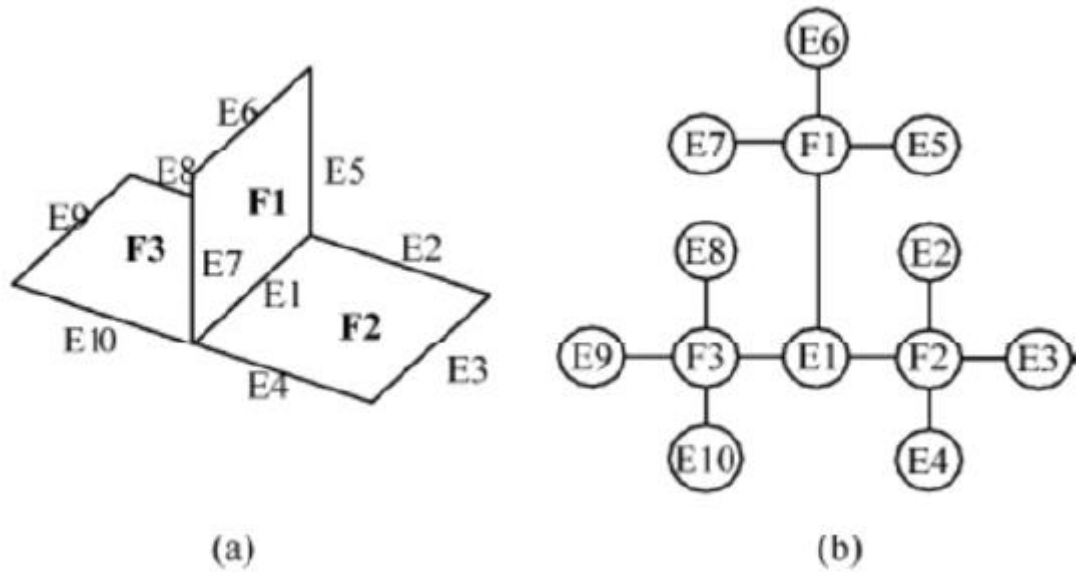


Fig.2.1 : Représentation des surfaces moyennes et des graphes d'adjacence [73]

Cette approche permet de définir les types d'entité (c'est-à-dire, trou, jonction en T, jonction en X, nervure) et de les représenter sous la forme des graphes d'adjacence. Les graphes sont ensuite convertis en graphes d'adjacence binaires pour effectuer des vérifications géométriques afin de valider et de catégoriser les entités reconnues. En menant des études de cas, ils prouvent que l'approche proposée génère avec succès un modèle solide de la pièce avec les entités reconnues. Cependant, leurs efforts de recherche sont limités car le module de décision de la fabricabilité de la pièce concerne seulement les types d'entités définies dans l'approche proposée.

Korosec et al. [74] proposent un modèle neuro-flou qui évalue la fabricabilité des entités de forme libre. Pour toutes les entités non prismatiques et non cylindriques, le modèle proposé définit de manière analytique des paramètres géométriques tels que le rapport surface/volume de l'entité, la somme de la courbure de la surface dans un plan et la somme des gradients sur les surfaces. Ces paramètres servent alors de variables pour l'évaluation de la fabricabilité. Cependant, leur cadre proposé est uniquement capable d'évaluer les opérations de fraisage.

2.3.2 *La visualisation et le retour d'information (feedback)*

Les travaux de recherche menée sur la fabricabilité à travers le retour d'information ou le feedback sont d'abord montrés dans les travaux de Mahajan [75]. L'auteur montre que l'étude de la fabricabilité est assurée par une approche simultanée. En utilisant la notion de feedback, l'auteur propose un système automatique qui fournit aux concepteurs des recommandations de reconception des pièces estampées.

Maropoulos et al. [76] reconnaissent l'importance et les avantages de fournir des feedback sur la fabricabilité pour les mesures de qualité au début de la conception. Ils proposent un système d'évaluation de la fabricabilité permettant de fournir des feedback de manière automatisée. En effet, le système développé aide les concepteurs à contrôler le coût et la qualité des produits et à prendre des décisions économiques relatives aux variations de tolérance. Le système proposé génère un ensemble de gammes d'usinage optimisées en proposant plusieurs alternatives de processus et de ressources de fabrication les plus appropriés à l'aide d'un algorithme de recuit simulé. Néanmoins, le travail proposé n'analyse que les processus d'usinage.

Gupta [77] exploite la planification de la fabrication sur un centre d'usinage 3-axes pour offrir aux concepteurs des possibilités de reconception. L'auteur développe un système qui permet de générer plusieurs gammes d'usinage. Pour choisir la meilleure solution, le système calcule le temps de production de chaque gamme et renvoie les informations générées aux concepteurs. Cependant, le système proposé ne fournit pas de feedback détaillé. Les concepteurs doivent déduire le problème de conception en comparant le temps estimé pour chaque opération de l'entité d'usinage correspondante. De plus, cette méthode n'est pas transférable à d'autres processus de fabrication, car le schéma de raisonnement de ce système ne peut pas être appliqué à un processus différent.

Ces efforts de recherche démontrent que le fait de fournir des feedback sur la fabricabilité au début de la conception peut réduire les itérations de conception les coûts et les délais de fabrication en éliminant rapidement les erreurs. Cependant, ces efforts ont tendance à se concentrer uniquement sur le contenu du feedback, et non pas sur le mode ou la structure de sa diffusion; par conséquent, les concepteurs qui reçoivent les feedback risquent de ne pas les comprendre en raison d'informations désordonnées ou incomplètes.

Un schéma de visualisation approprié peut aider à fournir les informations requises efficacement. Le but de la visualisation est de fournir des informations plus détaillées plutôt que des descriptions textuelles. L'analyse de la visibilité des processus d'usinage basée sur un algorithme de recherche de collision de chemin, illustrée à la figure 2.2, est une autre méthode récemment mise au point pour évaluer la fabricabilité [78].

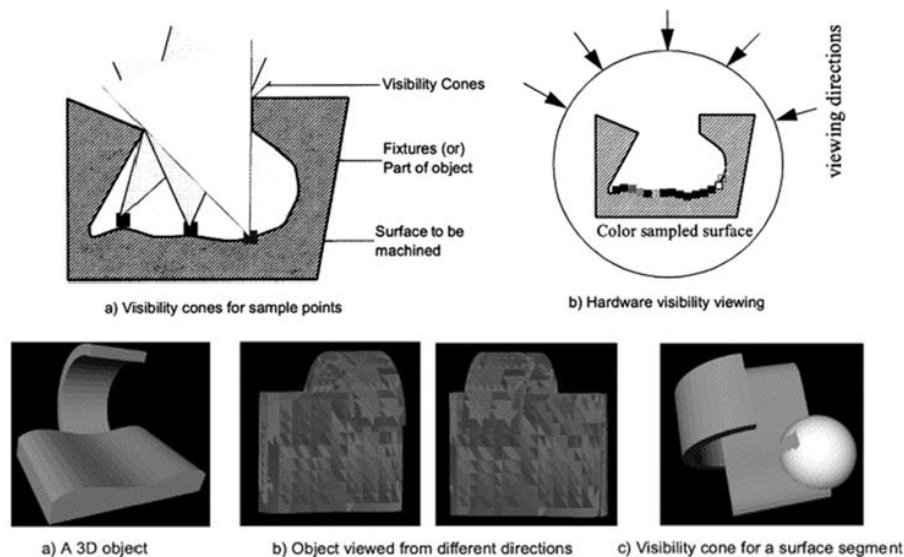


Fig.2.2 : L'analyse de la fabricabilité par la visualisation [78]

Dans le même contexte, une étude récente menée par Rangarajan et al. [79] intègre des analyses de la fabricabilité avec un jeu de couleurs. Les chercheurs se sont concentrés sur la capacité d'analyse de la fabricabilité des pièces en tôle en utilisant le langage (SADL) pour construire

un modèle sémantique. À l'aide du langage OWL (Web Ontology Language) et d'un moteur de raisonnement géométrique appelé Jena, ils créent une plate-forme de fabricabilité contenant une base de données de règles de conception et traitant les requêtes de conception générées à partir d'un fichier CAO. Pour le cas d'étude d'une pièce en tôle, ils montrent comment les informations géométriques de la pièce sont converties en un fichier OWL, puis comparées aux règles de conception dans l'environnement SADL. Une interface graphique Java est utilisée pour afficher les résultats de la vérification des règles avec un jeu de couleurs. Si la pièce n'est pas conforme aux règles prédéfinies alors la pièce apparaît en rouge. Dans le cas contraire, la pièce apparaît en vert lorsque toutes les règles de conception sont conformes. Cependant, l'étude ne peut pas être élargie au-delà d'un seul procédé de fabrication.

La réduction des boucles itératives au début de la conception nécessite un système d'ingénierie simultanée qui aide les concepteurs à évaluer l'ensemble des alternatives des conceptions possibles. Dans le même contexte, Chen et al. [80] proposent un système d'évaluation simultanée de la conception appelé (CONDENSE). Ces chercheurs envisagent une stratégie à deux couches pour déterminer les spécifications de la conception d'une part et pour choisir la meilleure conception qui optimise de l'autre part le coût, le temps et la fabricabilité. Le cadre proposé utilise une base de connaissances préalablement établie pour dériver des attributs de conception et pour traiter ces attributs dans un module de prise de décision. Pour mettre en œuvre le cadre proposé, ils créent des représentations visuelles d'une série de solutions de conception pour une tête de balle de golf en bois, comme le montre la figure 2.3. Le point remarquable de leur travail est que le cadre proposé génère d'une manière alternative plusieurs solutions de conception possibles pour que les concepteurs puissent évaluer les solutions de conception visuellement.

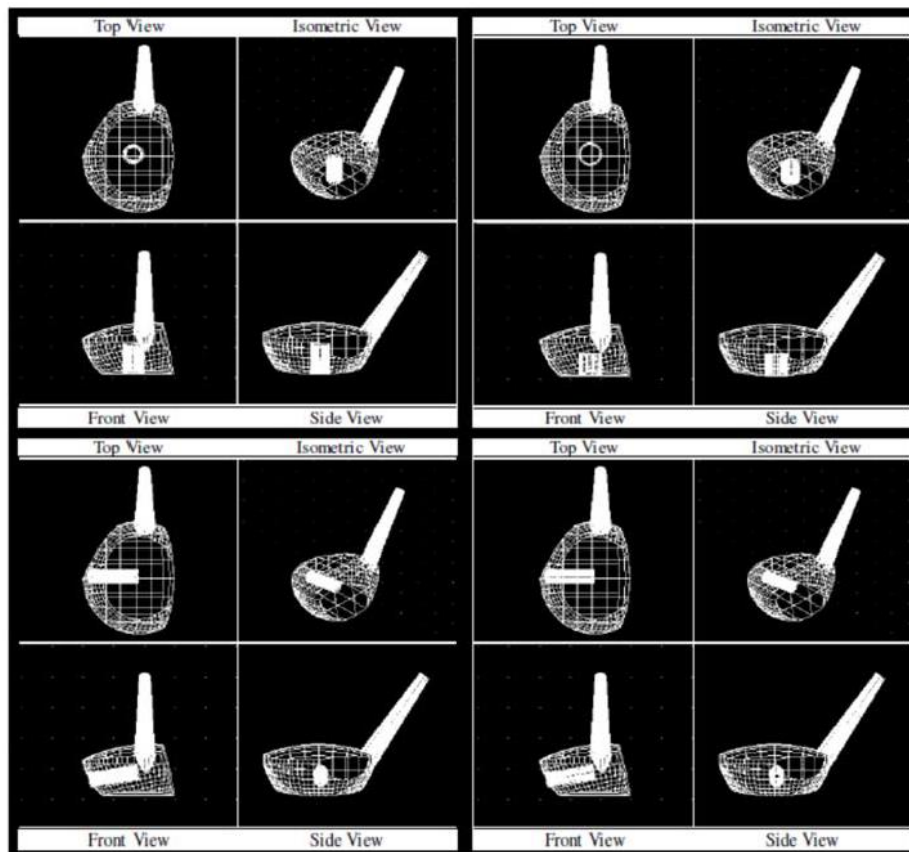


Fig.2.3 : La représentation des alternatives de conception générées par le système CONDENSE pour l'étude de la fabricabilité [80]

La visualisation des entités non conformes aux règles de la fabricabilité aide les concepteurs à reconnaître facilement les problèmes géométriques de la conception.

Dans la même perspective, Frank et son équipe de recherche ont développé un logiciel d'analyse de la fabricabilité appelé CNC-RP [81]. Ce logiciel effectue des analyses de la fabricabilité à travers quatre modules de visibilité, d'accessibilité des machines-outils, d'usinabilité et de complexité de montage d'usinage pour une fraiseuse 3-axes. Chaque analyse inspecte une conception et donne un calcul de temps d'usinage avec des résultats d'analyse visualisés à l'aide d'un schéma de couleurs, comme indiqué dans la figure 2.4.

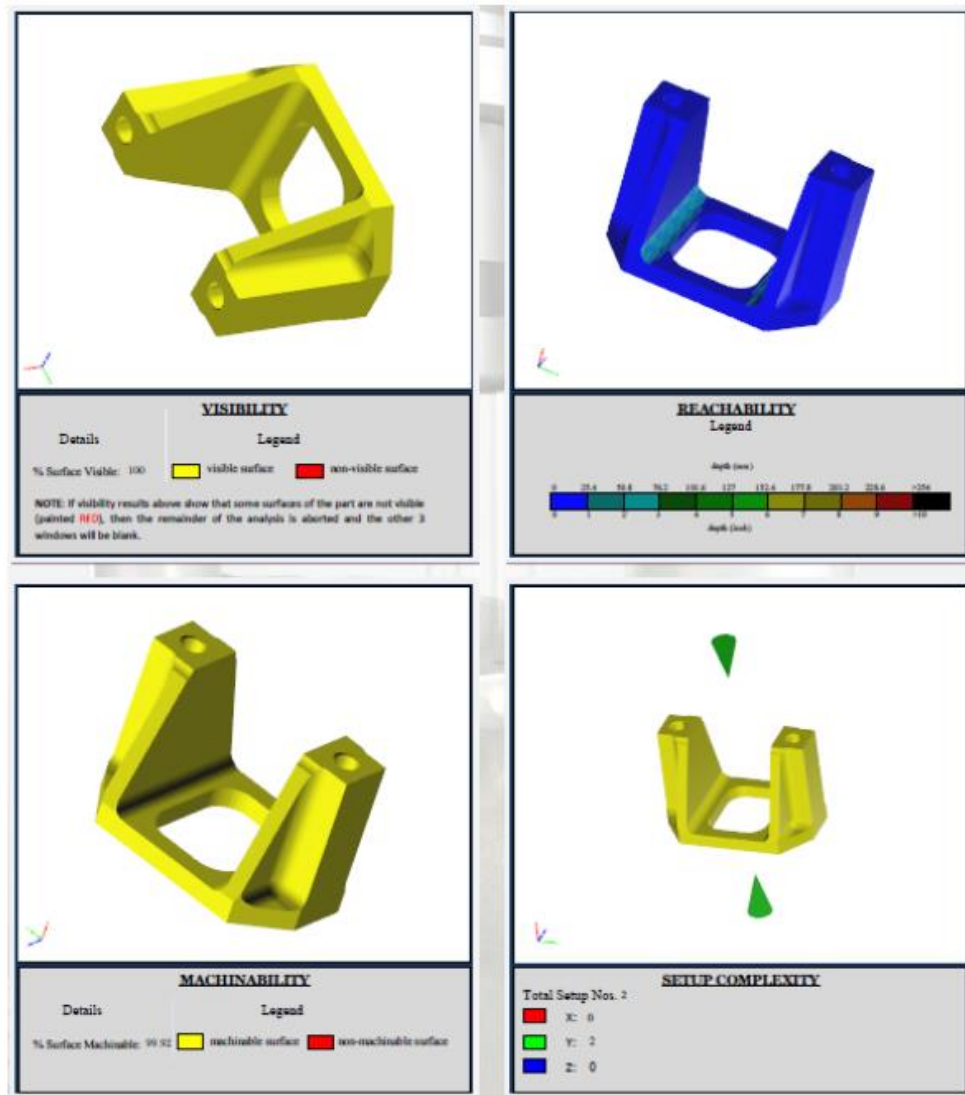


Fig.2.4 : Les modules d'analyse de la fabricabilité du logiciel CNC-RP [81]

Le logiciel CNC-RP repose sur la technologie 3DIF qui permet de représenter les 4 modules d'analyse. Cette technologie comporte :

1. L'analyse de visibilité qui permet d'examiner la conception de la pièce. Si une pièce a une section creuse à l'intérieur qui n'est pas visible sous aucun angle, l'analyse renvoie un message d'erreur et la pièce ne peut plus être analysée. En d'autres termes, toutes les sections d'une pièce doivent être visibles pour pouvoir être usinées.

2. L'analyse de l'accessibilité qui permet de mesurer l'écart entre les facettes de la pièce. Une échelle de couleurs est préparée du bleu au noir en fonction de la distance renvoyée (0 pouce à 10 pouces). Si l'écart augmente, la longueur de l'outil doit être plus longue et risque davantage de se casser. En conséquence, la vitesse d'usinage doit être réduite, ce qui augmente les temps et les coûts de traitement.

3. L'analyse d'usinabilité qui détermine si l'outil peut atteindre chaque facette, ce qui dépend du diamètre de l'outil. Dès que le diamètre de l'outil augmente, l'analyse renvoie davantage de facettes rouges, ce qui signifie qu'elles ne sont pas fabricables.

4. l'analyse de la complexité du montage qui renvoie tous les axes requis sous forme de cônes nécessaires pour garantir l'accessibilité de l'outil.

Ce logiciel offre des moyens efficaces d'afficher les résultats de l'analyse afin que les concepteurs puissent identifier rapidement les problèmes liés à la fabricabilité de la pièce. Il permet également de manipuler les angles de vision de chaque pièce afin que toutes les surfaces existantes puissent être examinées de manière approfondie. Pour l'analyse de l'accessibilité, un schéma de couleurs est utilisé pour représenter les profondeurs relatives des surfaces. Les surfaces difficilement accessibles par un outil d'usinage sont marquées par des couleurs chaudes. Pour l'analyse d'usinabilité, les surfaces qui ne peuvent pas être usinées sont représentées en rouge. Grâce à la visualisation du logiciel CNC RP, les feedback peuvent être mieux interprétées que les descriptions textuelles.

2.3.3 Les directives de la conception pour la fabrication (DFM)

Les directives de la conception pour la fabrication (DFM) peuvent servir de bases de données pour évaluer la fabricabilité à un stade précoce de la conception. En effet, Lazaro et al. [82] proposent un système de prise de décision basé sur les règles géométriques pour les pièces en

tôle. Le cadre proposé relie les données d'un sous-système de CAO à un système basé sur les connaissances appelé Sheet Metal Advisor afin d'examiner la fabricabilité de chaque entité. Ce système permet de proposer des suggestions de reconception.

Venkatachalam et al. [83] ont développé une approche basée sur les directives de la conception pour la fabrication au début de la phase de conception. L'avantage de cette approche est d'intégrer les contraintes de conception avec les paramètres de fabrication. En effet, les auteurs proposent un système expert basé sur des règles de fabrication qui permet d'une part de sélectionner les processus primaires et secondaires convenables aux données de conception et d'autre part d'estimer les coûts pour les processus sélectionnés. Les processus primaires intègrent le moulage et le forgeage. Pour les processus secondaires, le système proposé prend en compte les opérations de fraisage et de perçage. En utilisant le critère DFM, les chercheurs ont réussi à créer des mesures de fabricabilité pour un ensemble de conceptions.

Prasad et al. [84] présentent également une approche basée sur les directives de la conception pour la fabrication d'équipements médicaux. L'approche proposée commence par évaluer la fabricabilité des composants constituant un système d'équipement médicale. La deuxième étape consiste à suggérer des recommandations de reconception de ces composants afin d'éliminer les étapes de traitement inutiles et d'optimiser la production. Par conséquent, les auteurs parviennent à réduire le coût total de 57% et le temps de production de 77%.

Molcho et al. [85] ont développé un outil d'analyse de la fabricabilité assistée par ordinateur (CAMA) pour capturer les connaissances de fabrication et pour évaluer la conception. Inspirés par les méthodologies de conception pour X (DFX), les auteurs visent à réduire le nombre d'itérations et les flux d'informations multiples entre concepteurs et experts. L'outil proposé comprend des modules pour : gérer des modèles CAO à travers l'outil PTC Pro Engineer,

collecter les règles de conception avec une architecture syntaxique bien définie, traduire de manière hiérarchique les règles collectées en codes C ++, et enfin évaluer la fabricabilité du modèle CAO à travers la base de règles. La structure de l'outil CAMA est illustrée à la figure 2.5. Cet outil permet de construire un cadre complet pour la gestion des directives de la conception pour la fabrication. Cependant, la portée de logiciel est limitée aux processus d'usinage.

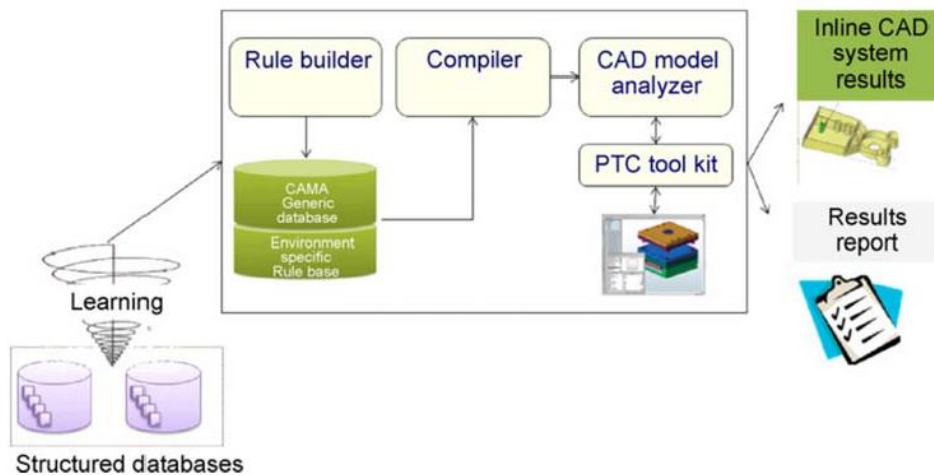


Fig.2.5 : Les composants de l'outil CAMA pour l'analyse de la fabricabilité [85]

Bidkar et al. [86] proposent un cadre d'évaluation automatisé de la fabricabilité pour les pièces moulées par injection en utilisant un algorithme basé sur les directives de la conception pour la fabrication, comme illustré à la figure 2.6. Le cadre proposé se compose de deux couches. Premièrement, les auteurs définissent un algorithme d'analyse de la fabricabilité qui examine globalement et systématiquement les propriétés d'une pièce. Pour chaque étape de l'algorithme, les entités géométriques sont reconnues et représentées dans un format binaire dans le programme C ++ et examinées individuellement pour déterminer leur fabricabilité. Deuxièmement, les entités individuelles capturées sont rassemblées, puis la fabricabilité d'une conception est déduite en suivant l'algorithme proposé. Même si le cadre proposé fournit des

analyses approfondies des pièces moulées par injection, l'architecture de l'algorithme DFM nécessite un codage complexe et ne peut pas être facilement transférée à d'autres processus.

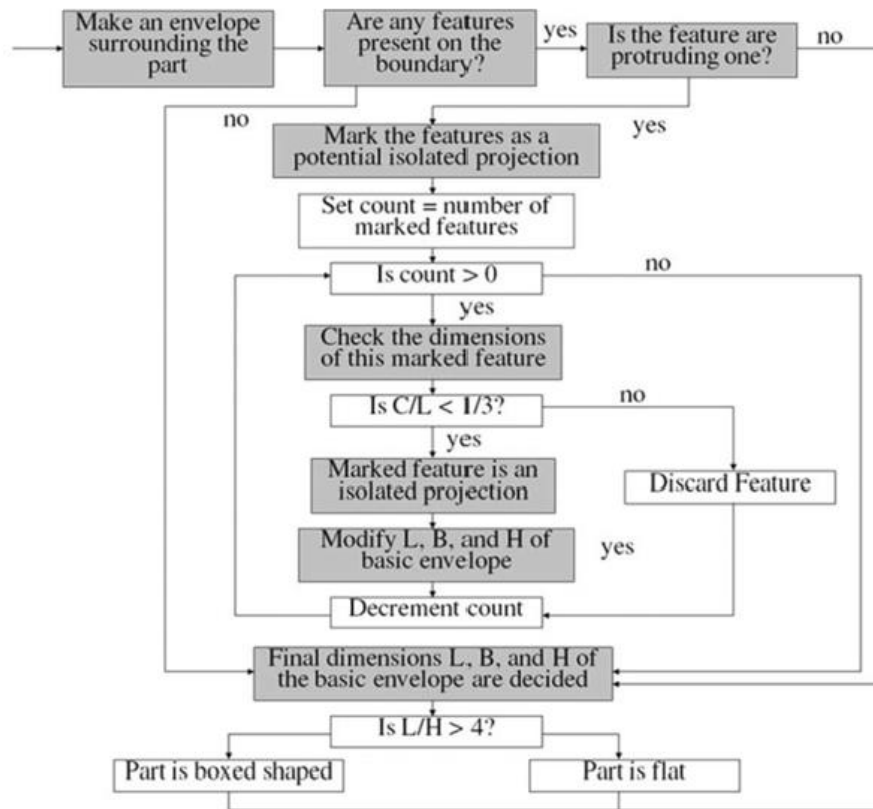


Fig.2.6 : Algorithme basé sur les directives du (DFM) pour l'étude de la fabricabilité des pièces moulées par injection [86]

Jacob et al. [87] proposent une méthode d'évaluation de la fabricabilité en comparant les entités géométriques d'une pièce à des règles géométriques constituées à travers une base de données textuelle. L'organigramme de cette méthode est illustré dans la figure 2.7. Premièrement, le modèle de la pièce et les règles géométriques liés au processus de rectification sont codés dans le système d'évaluation proposé. Ensuite, lorsque le système reconnaît toutes les entités de la pièce, les règles de conception correspondantes sont appelées pour générer des décisions de fabricabilité. Malgré l'effort déployé pour développer cette méthode, la collecte de ces

directives de conception et leur conversion en code prend du temps et nécessite un codage complexe pour ajouter ou modifier une règle géométrique.

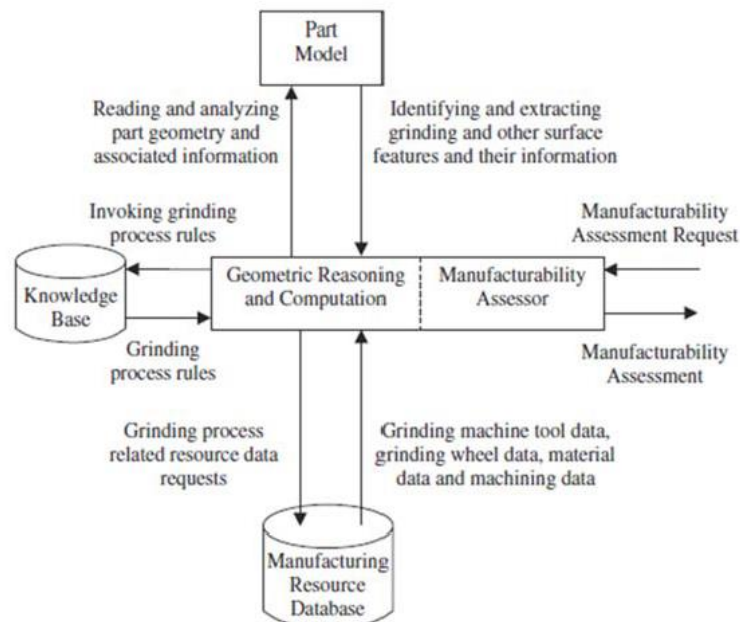


Fig.2.7 : Modèle d'analyse de la fabricabilité pour le processus de rectification [87]

Amalnik et al. [88] adoptent une approche orientée objet (OOT) pour étudier la fabricabilité d'une pièce dans un environnement d'ingénierie simultanée. L'objectif de cette approche est d'obtenir des mesures de qualité pour la prise de décision, y compris la durée et le coût du cycle d'usinage, le taux d'enlèvement de matière et l'efficacité de l'opération. Afin d'estimer ces mesures, neuf classes différentes des entités de conception sont intégrées dans un système intelligent basé sur la connaissance (iKBS). L'enquête sur l'efficacité de la fabrication est ensuite effectuée en prenant en compte les relations entre les entités de conception et les entités de classe.

Il ressort clairement des discussions qui précèdent que les directives de conception pour la fabrication peuvent jouer un rôle dans la vérification de la fabricabilité d'une conception ou d'un modèle, ainsi que dans la suggestion des améliorations d'entités géométriques.

2.4. Tableau comparatif sur les méthodes intégrant la fabricabilité

Dans les sections précédentes, nous avons défini la fabricabilité et son rôle à réduire les boucles itératives dès la phase amont de conception. Dans cette partie, nous allons comparer les travaux étudiés dans le but de montrer les avantages et les inconvénients de chaque méthode.

Critères	Les travaux correspondants	Avantages et inconvénients
Reconnaissance d'entités	. Développement d'un système pour récupérer les données de l'entité d'usinage à travers le toolkit de Pro/Engineer afin de développer des gammes d'usinage [69]. . Utilisation d'un outil (IF²) pour extraire les entités selon les standards AP 203 et AP 224 afin d'optimiser le coût de fabrication [70]. . Développement d'un système expert qui permet de convertir des données de bas niveau extraites des entités géométriques en information d'usinage comme les séquences d'usinage et les types d'outils à utiliser [71]. . Développement d'une méthode d'extraction des entités géométriques pour une approche basée sur les entités du processus de moulage par injection à partir d'une conception arbitraire [72]. . Développement d'une autre méthode d'extraction des entités de moulage par injection à travers les graphes d'adjacences afin de reconnaître les entités [73]. . Développement d'un modèle neuro-flou qui évalue la possibilité de fabrication des entités de forme libre [74].	Ces travaux de recherche intégrant la reconnaissance des entités géométrique de la pièce de quelques processus permettent de fournir de bons résultats pour analyser la fabricabilité. Parmi, les données récupérées, on trouve les dimensions des entités et les tolérances géométriques. Néanmoins, pour une conception arbitraire ou les entités ne sont pas bien définies selon les protocoles d'application AP 224 ou AP203 ou bien si l'entité n'est pas libre, c'est-à-dire qu'il s'agit d'une entité interagissante, ceci peut retarder l'analyse de la fabricabilité et augmenter ainsi le temps et le coût de développement. un autre inconvénient c'est que la plupart de ces travaux traitent seulement un seul processus de fabrication qui est dans la plupart des cas l'usinage.
Visualisation et retour d'information (feedback)	. Utilisation de la notion du feedback pour fournir aux concepteurs des recommandations de reconception des pièces estampées [75]. . Développement d'un système d'évaluation de la fabricabilité permettant de fournir des feedback de manière automatisée [76]. . Développement d'une approche qui permet aux concepteurs la possibilité de reconception en exploitant la planification de la fabrication de haut niveau sur un centre d'usinage 3-axes [77].	La méthode basée sur la visualisation et le retour d'information peut aider à fournir les informations requises efficacement surtout pour des concepteurs ayant un niveau de conception pour la fabrication faible ou moyen. Ceci permet d'évaluer la fabricabilité en terme de visibilité, d'accessibilité des

	. Analyse de visibilité des processus d'usinage basée sur un algorithme de recherche de collision de chemin pour mesurer la fabricabilité de la pièce [78]. . Développement d'un modèle sémantique pour l'analyse de la fabricabilité en se basant sur un jeu de couleur [79]. Génération de plusieurs solutions de conception à travers un système d'évaluation simultanée de la conception (CONDENSE) pour évaluer la fabricabilité visuellement [80]. . Développement d'un logiciel d'analyse de la fabricabilité appelé CNC-RP en terme de visibilité, d'accessibilité des machines-outils, d'usinabilité et de complexité de montage pour une fraiseuse 3-axes [81].	machines et de complexité de configuration. Cependant, l'étude ne peut pas être élargie au-delà d'un seul procédé de fabrication.
Les directives de la conception pour la fabrication (DFM)	. Intégration d'un sous-système de CAO avec un système basé sur les directives DFM appelé Sheet Metal Advisor afin d'examiner la possibilité de fabricabilité de chaque entité d'une pièce en tôle [82]. . Développement d'un système expert basé sur des règles de fabrication qui permet d'une part de sélectionner les processus primaires et secondaires convenables aux données de conception et d'autre part d'estimer les coûts pour les processus sélectionnés [83]. . Développement d'une approche basée sur DFM dans la fabrication d'équipements médicaux afin de réduire les coûts et les temps de production [84]. . Développement d'un outil d'analyse de la fabricabilité assistée par ordinateur (CAMA) pour implémenter les connaissances en matière de fabricabilité [85]. . Proposition d'un cadre d'évaluation automatisé de la fabricabilité pour les pièces moulées par injection en utilisant un algorithme DFM [86]. . Développement d'un modèle pour évaluer les entités de fabrication liées au processus de rectification [87]. . Adoption d'une approche orientée objet (OOT) qui intègre un système intelligent basé sur la	Les directives de la conception pour la fabrication (DFM) ont été appliquées dans plusieurs domaines notamment dans la fabrication des équipements médicaux. l'avantage de ce critère est que chaque recommandation sert de critère heuristique pour déterminer la fabricabilité d'un processus. Ces directives peuvent aussi suggérer des recommandations d'amélioration des entités géométriques. ces directives peuvent ainsi servir comme base de données pour guider les concepteurs lors de la conception d'un nouveau système. Le problème majeur avec ces travaux c'est que les règles de fabrication sont appliquées dans la plupart des cas pour le processus d'usinage et nécessitent une collecte d'une quantité énorme d'information pour constituer la base de connaissance.

	connaissance (iKBS) pour obtenir des mesures de qualité y compris la durée et le coût du cycle d'usinage, le taux d'enlèvement de matière et l'efficacité de l'opération [88].	
--	--	--

Tab.1 : Tableau comparatif sur les méthodes intégrant la fabricabilité

2.5.Contribution de notre travail pour l'étude de la fabricabilité

L'étude de travaux intégrant la fabricabilité dans la section précédente présente plusieurs lacunes :

- Utilisation de plusieurs types de standards pour l'échange de données entre ingénieurs et experts ce qui retarde le temps de développement du produit.
- Le développement d'un système ou d'un logiciel d'analyse de la fabricabilité est coûteux et nécessite un codage complexe pour extraire les entités de fabrication et les comparer aux règles géométriques. En plus, lorsqu'il s'agit d'une forme complexe de la pièce, le système peut mal interpréter les informations de conception.
- Le cadre de développement du produit ne permet pas aux parties prenantes d'échanger leurs idées et leurs domaines d'analyse afin d'optimiser la conception de la pièce.
- L'analyse de la fabricabilité dans ces travaux est basée seulement sur un seul procédé de fabrication qui est dans la plupart des cas l'usinage. D'autres procédés peuvent assurer un bon rapport qualité/prix.
- Les contraintes liées aux ressources de fabrication disponibles sur les sites de production ne sont pas considérées lors de l'étude de la fabricabilité.

Pour surmonter ces lacunes, l'originalité de notre travail consiste à développer une approche qui intègre l'ingénierie systèmes basée sur les modèles (MBSE) avec les principes de l'approche SBCE afin de réduire les boucles itératives entre les concepteurs inexpérimentés, les fabricants et les experts. D'autres parties prenantes participent également au développement

du produit afin d'assurer un environnement d'ingénierie simultanée. Cette approche offre un grand potentiel pour gérer les aspects de fabricabilité et de fiabilité, en raison de la capacité à développer des connaissances en parallèle, en particulier pour les technologies avancées et non matures. L'avantage d'utilisation des principes du SBCE est d'offrir des possibilités de communication entre les parties prenantes au début de la phase de conception. SysML est utilisé comme langage descriptif pour représenter les différentes phases de cette approche. Pour assurer la communication entre l'ingénieur système et les participants à la conception, deux type de flux sont présentés dans cette approche comme le montre la figure 2.8 :

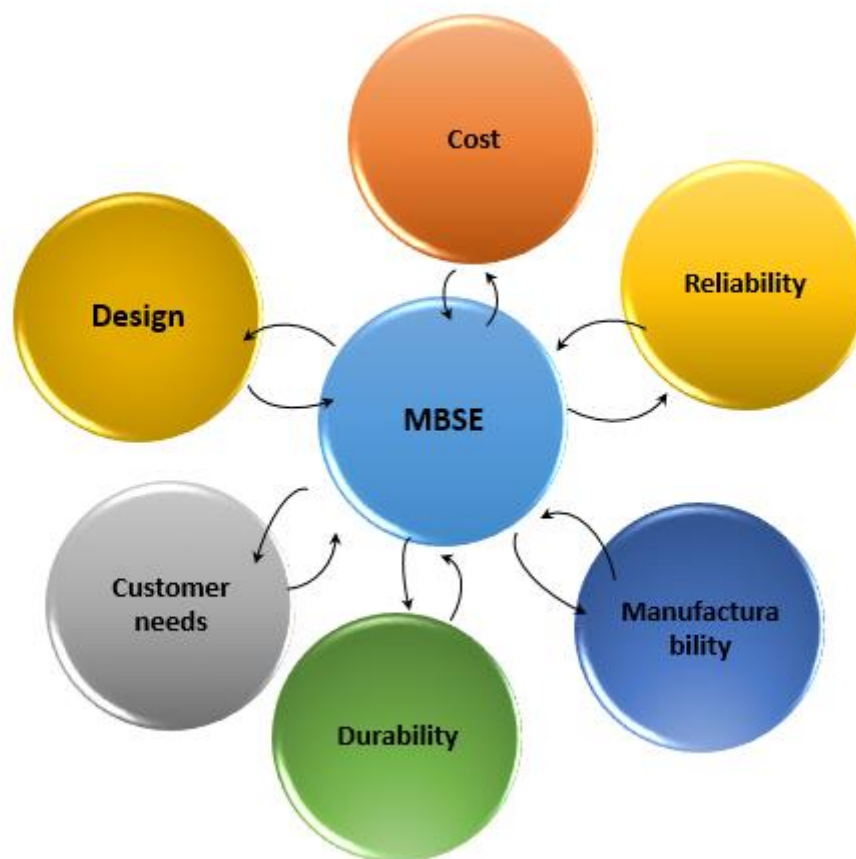


Fig.2.8 : Les types de flux échangés entre l'ingénieur système et les participants à la conception

- Un flux descendant qui représente l'échange d'informations issu de l'ingénieur système vers les différents spécialistes pour mener des analyses spécifiques qui sont dans leurs domaines,

- Un flux ascendant qui sert de feedback pour aider l'ing nieur syst me lors de l'analyse des compromis.

L'objectif de cette approche est d'optimiser en premier lieu le choix de la meilleure architecture syst me en menant une  tude de la fiabilit  sur les alternatives de conception des syst mes m catroniques et d'optimiser en deuxi me lieu le choix du processus de fabrication qui minimise le co t et le temps de fabrication. Pour cela nous avons d velopp  deux mod les   travers les diagrammes de SysML qui se basent sur les principes du SBCE. Le premier mod le intitul  « Model-Based for robustness analysis » est d velopp  dans le chapitre suivant. Pour l' tude de la fabricabilit , nous avons d velopp  un autre mod le intitul  « Model-Based for manufacturability analysis » dans le chapitre 4. Pour cartographier l'espace des solutions, les solutions de conception sont remplac es par un ensemble de processus de fabrication qui sont disponibles dans l'atelier de fabrication de l'entreprise. Ensuite pour r duire le nombre d'alternatives, nous allons adopter des filtres bas s sur les directives de la conception pour la fabrication afin de v rifier la capacit  de chaque processus. Le principe d'int gration par intersection de l'approche SBCE est remplac  par l'utilisation d'un nombre minimal de r gles g om triques qui sont partag es entre les diff rents processus de fabrication afin de converger vers un nombre limit  de solutions optimales. En fin, pour optimiser le choix des processus, une  tude des compromis est men e afin d'optimiser le co t, le temps et la qualit . L'adoption d'une approche bas e sur les ensembles dans le domaine de l'analyse de la fabricabilit  peut aider   r duire les contraintes de fabrication qui d terminent la faisabilit  de la fabrication tout en maintenant l'efficacit  de l'analyse de la fabricabilit .

Chapitre 3

Développement du modèle MBRA pour le choix de la meilleure architecture système

3.1.	Introduction.....	59
3.2.	L'approche proposée.....	60
3.3.	Implémentation de l'algorithme	69
3.4.	Etude de cas d'un compresseur électrique	71
3.5.	Comparaison avec des travaux similaires.....	81
3.6.	Discussion	82

3.1. Introduction

Dans ce chapitre, nous proposons une approche qui repose sur l'application d'un nouveau modèle intitulé «Model Based for Robustness Analysis» (MBRA) qui est utilisé pour le choix de l'architecture d'un système mécatronique. En effet, ce modèle intègre l'MBSE avec les principes du SBCE dans les phases de conception préliminaires. Il a pour objectif de filtrer les solutions faibles ou coûteuses au fil du temps et d'assister l'ingénieur système lors de l'adoption d'une étude de compromis. Premièrement, les besoins du client, ainsi que les aspects fonctionnels et structurels du système, sont définis à l'aide des diagrammes SysML. Ensuite, l'application d'un premier filtre pour tester les sous-systèmes est développée parallèlement avec l'étape qui se base sur le 3^{ème} principe du SBCE « Développement d'un ensemble de concept ». Pour appliquer le 4^{ème} principe du SBCE « convergence des concepts », un algorithme est développé après la phase d'intégration des systèmes conçus afin de réduire l'espace de conception et, finalement, une analyse de compromis est effectuée pour sélectionner la meilleure architecture système. La première contribution de ce chapitre est la création de nouveaux stéréotypes de flux de composition et d'affectation entre l'activité principale et les sous-systèmes ou entre sous-systèmes et composants dans un diagramme de profil SysML. La deuxième contribution est la mise en œuvre d'algorithmes permettant d'analyser la fiabilité du système et de converger vers la meilleure solution qui répond au mieux aux besoins des clients et des parties prenantes.

Dans la section suivante "L'approche proposée", une approche systématique basée sur les diagrammes SysML et les principes SBCE est développée pour automatiser la cartographie de l'espace de conception et pour réduire les délais et les coûts de développement de produits. La section 3 "Implémentation d'algorithme", présente un algorithme qui a été implémenté avec l'outil Gephi et le langage Python pour effectuer l'analyse de la fiabilité. La section 4 est dédiée

à l'étude de cas d'un compresseur électrique automobile pour illustrer les différentes étapes de cette approche d'ingénierie simultanée. La section 5 est une étude comparative entre ce modèle et les travaux réalisés pour le développement des produits complexes. Enfin, La dernière section " Discussion et conclusion " présente les résultats de ce modèle.

3.2. L'approche proposée

Le but de cette approche est de fournir un modèle commun (MBRA) aux différentes parties prenantes lors des premières phases de conception et de partager leurs idées et leur analyse de domaine spécifique avec l'ingénieur système en tant que responsable de la gestion d'une plateforme commune qui traite plusieurs aspects du système. Cela se fait à travers les diagrammes SysML. En fait, cet outil descriptif permet de développer un environnement d'ingénierie simultanée dans lequel des équipes multidisciplinaires travaillent ensemble depuis le stade des exigences client jusqu'à l'identification de l'ensemble des différentes solutions techniques. Cette approche descendante est développée parallèlement avec les principes de l'approche SBCE. L'utilisation de ces principes a pour objectif de réduire les délais et le nombre de modifications tardives. En outre, l'SBCE permet de développer des connaissances sur plusieurs concepts de conception en ajoutant des informations issues des tests et de l'analyse de la robustesse du système. Sur la base de ces informations, les solutions faibles ou coûteuses sont filtrées successivement de l'espace de conception jusqu'à converger vers la meilleure solution.

La figure 3.1 illustre les phases de l'approche proposée :

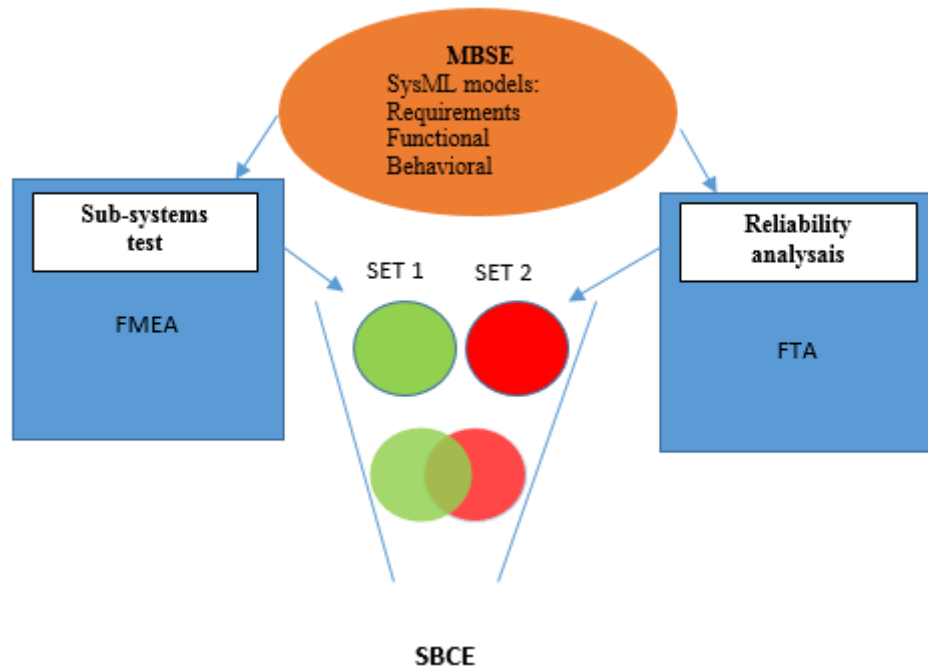


Fig.3.1 : Le modèle MBRA

La première phase de conception de cette approche commence par la définition et la prise en compte des besoins du client et des parties prenantes internes ainsi que la fonctionnalité du système au début du processus de développement. Cette phase vise à bien comprendre les valeurs du client, ce qui est une condition préalable essentielle à la définition des objectifs du système et à l'évaluation de la rationalité des conceptions alternatives. Cette phase correspond au premier principe du SBCE « Recherche de valeurs ». Les exigences système du niveau supérieur sont décomposées en exigences plus détaillées, puis structurées dans le diagramme d'exigences de SysML. Les valeurs du client sont capturées et classées comme des attributs de valeurs clé (Key Value Attributs).

Pour définir les valeurs les plus importantes, nous adaptons un processus d'hiérarchie analytique (AHP), qui est un outil efficace pour traiter des décisions complexes, et permet donc de déterminer les exigences relatives aux attributs de valeurs clé (KVA).

Pour cartographier l'espace de conception, les parties prenantes ou les équipes de sous-systèmes définissent la portée des travaux de conception requise, ainsi que les options / régions de conception réalisables. En fait, cette activité traduit les exigences fonctionnelles que le système doit exécuter. En effet, pour définir l'architecture fonctionnelle du système, un processus d'itération entre les fonctions dérivées d'exigences détaillées et les exigences allouées à des fonctions est une étape importante pour fournir un cadre structuré de développement. Cette phase est développée simultanément avec le 2^{ème} principe de l'SBCE « cartographier l'espace de conception ». L'architecture fonctionnelle est modélisée à l'aide du diagramme d'activité de SysML. Au cours de cette phase de conception, les flux d'entrée / sortie sont clarifiés et les actions de base sont présentées. Le comportement du système est décrit à l'aide des diagrammes machine d'états de SysML. L'ensemble des événements, états et actions possibles est ensuite identifié, structuré et tracé vers les fonctions et les exigences du système.

Pour modéliser les performances du système et pour permettre la prise de décisions de manière efficace, une architecture d'allocation est nécessaire pour intégrer la décomposition des exigences avec l'architecture fonctionnelle et physique. Dans cette phase de conception, chaque fonction du système est attribuée à un ou plusieurs sous-systèmes au moyen d'une matrice d'allocation. L'utilisation de cette architecture d'allocation présente l'avantage de décrire de manière complète la conception du système et de répondre aux besoins des parties prenantes. Dans une étape avancée du processus d'ingénierie systèmes, l'architecture physique est définie à l'aide du diagramme de définition de blocs de SysML (BDD). Les sous-systèmes peuvent ensuite être identifiés et les flux échangés entre sous-systèmes peuvent être décrits à l'aide du diagramme de blocs internes de SysML (IBD). La décomposition des sous-systèmes permet de définir les solutions techniques des composants et de leurs attributs.

Dans cette phase de conception, le nombre de solutions possibles peut être très élevé et les décisions concernant la solution finale sont difficiles à prendre car elles nécessitent des informations plus détaillées émanant des spécialistes de chaque discipline tels que la fiabilité, la durabilité, la performance et le coût. Pour résoudre ce problème, nous créons un diagramme de profil dans SysML, comme illustré dans la figure 3.2, contenant les informations nécessaires à chaque spécialiste.

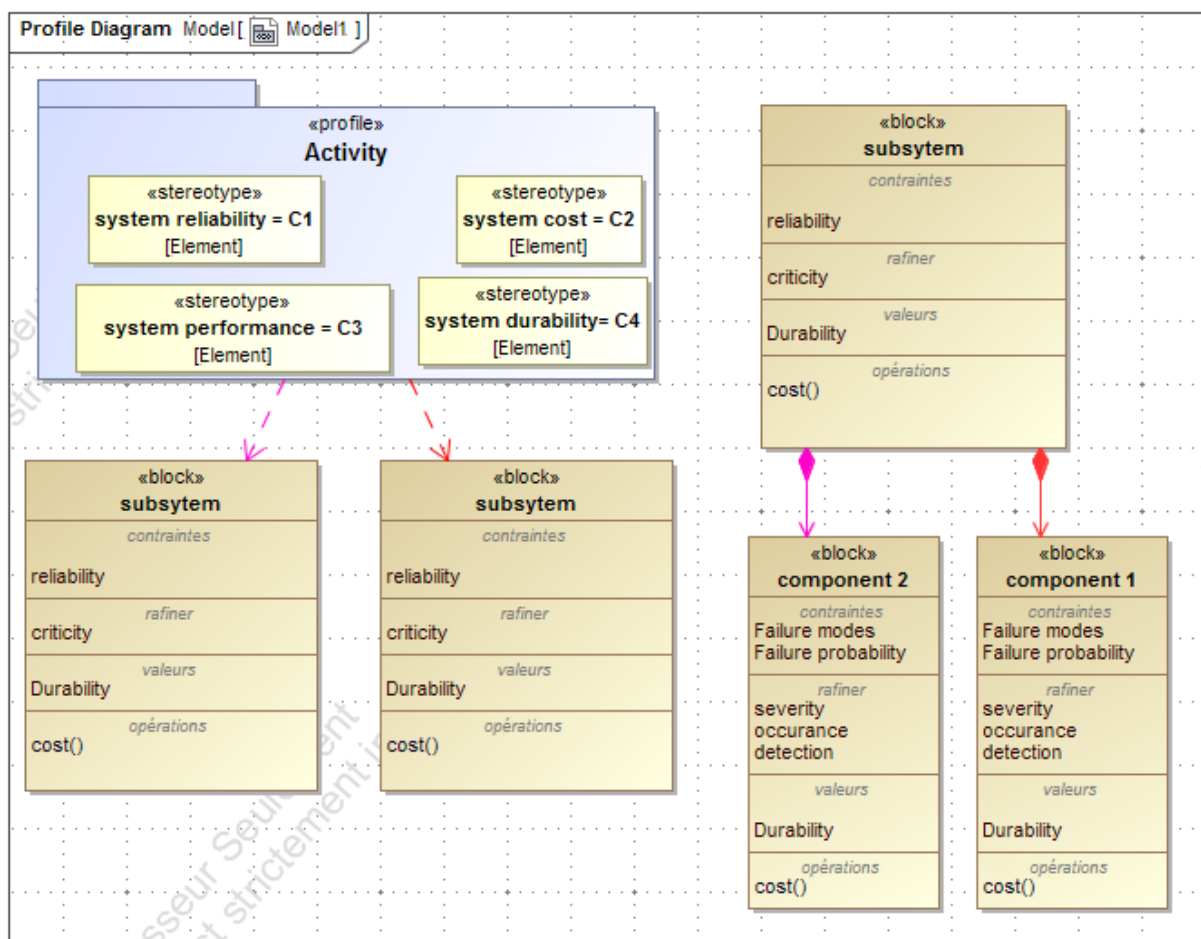


Fig.3.2 : Le diagramme de profil SysML développé pour l'étude de la fiabilité

Par exemple, pour tester la fiabilité des sous-systèmes, nous introduisons des attributs dans un compartiment spécifique du bloc, tels que le mode et la cause de défaillance des composants. Chaque compartiment du bloc des composants correspond à des attributs nécessaires à l'analyse d'un domaine spécifique. Une autre nouveauté de ce diagramme de profil est la prise en compte

des flux de composition et d'allocation pour un ensemble de solutions alternatives. Étant donné que les diagrammes de SysML ne permettent de représenter qu'une solution à partir de l'espace de conception, alors l'ajout d'autres stéréotypes de flux entre sous-systèmes ou entre des sous-systèmes et une activité est essentiel pour implémenter l'aspect des solutions basé sur les ensembles. Dans la figure 3.2, le bloc des composants est stéréotypé, il contient 4 compartiments correspondant aux analyses de la fiabilité, de la criticité, de la durabilité et du coût. Pour les stéréotypes de flux, le flux rouge entre une activité et un bloc ou entre deux blocs correspond à la porte « OU », ce qui signifie qu'il existe plusieurs solutions possibles alors que le flux violet correspond à la porte « ET », ce qui signifie qu'il n'y a qu'une seule alternative. L'analyse du coût, de la durabilité et de la performance des systèmes conçus serviront comme valeurs clé pour l'analyse des compromis lors du choix de la meilleure architecture système.

Pour modéliser l'espace de conception et pour cartographier la structure des diagrammes de SysML de cette approche, nous utilisons une formalisation des exigences, des sous-systèmes, des activités et des composants sous forme d'une structure mathématique comme suit :

$F = \{R, A, S, C, V, K, G\}$ où:

- $R = \{r_1, \dots, r_n\}$ est l'ensemble des exigences que le système doit respecter.
- $A = \{a_1, \dots, a_m\}$ est l'ensemble des activités dérivées que le système doit exécuter.
- $S = \{s_1, \dots, s_p\}$ est l'ensemble des sous-systèmes alternatifs alloués à chaque activité.
- $C = \{c_1, \dots, c_z\}$ est l'ensemble des composants qui constituent le sous-système.
- $V \subseteq (R \times A)$ est l'ensemble des relations entre exigences et activités où, pour chaque exigence détaillée $r \in R$, il existe au moins une activité $a \in A$ telle que $V(r, a)$. Cette relation décrit les combinaisons possibles d'activités dérivées d'exigences détaillées.

- $K (A \times C)$ est l'ensemble des relations entre sous-système et activités où, pour chaque sous-système $s \in S$, il existe au moins une activité $a \in A$ telle que $K (s, a)$. Cette relation traduit les relations d'allocations entre les activités et les sous-systèmes.
- $G \subseteq (S \times C)$ est l'ensemble des relations entre sous-systèmes et composants où, pour chaque sous-système $s \in S$, il existe au moins un composant $c \in C$ tel que $G (s, c)$. Cette relation définit la combinaison possible de composition entre sous-systèmes et composants.

Pour tester et explorer les ensembles de sous-systèmes, nous utilisons l'outil AMDEC qui est largement utilisé par les normes liées à l'industrie automobiles comme la norme IATF 16949. Dans cette approche, l'outil AMDEC a pour objectif de tester la qualité de conception des sous-systèmes et d'éliminer certaines solutions critiques dans l'environnement SysML.

En fait, dans l'industrie automobile, les besoins des clients et les exigences de la norme IATF 16949 doivent être alignés afin de définir les objectifs du système avec la qualité et les performances requises. Nous commençons par rechercher les modes de défaillance courants des différents composants, puis nous définissons les causes racines de chaque mode de défaillance. Ensuite, l'effet de chaque mode de défaillance d'un composant est considéré comme cause de défaillance du sous-système. Étant donné que l'AMDEC est utilisé par de nombreuses industries manufacturières telles que les industries de l'automobile et de l'aérospatiale, une norme commune pour les systèmes mécatroniques doit être utilisée pour définir les échelles d'évaluation de gravité, d'occurrence et de détection. La norme la plus largement utilisée est la norme MIL-STD-1629A, établie par le ministère de la Défense des États-Unis [89]. Avec plus de 30 ans d'utilisation et de développement, cette norme a été utilisée dans de nombreux secteurs pour l'analyse des défaillances. En raison de la complexité et de la criticité des systèmes mécatroniques, elle constitue une base fiable pour effectuer des

tests AMDEC sur divers systèmes. Elle contient également des formules permettant de prédire les taux de défaillance des systèmes complexes, dont les coefficients sont basés sur des tests de durée de vie.

La gravité (S) indique la gravité des effets d'un mode de défaillance, l'occurrence (O) indique la probabilité qu'une défaillance se produise et la détection (D) mesure la visibilité d'une défaillance ce qui traduit l'attitude d'un mode de défaillance à être identifié par des commandes ou par des inspections. Chaque paramètre est corrélé à une plage de scores (minimum de 1 à maximum 4 dans ce travail). Pour calculer et comparer les criticités du sous-système, nous commençons par calculer les criticités des composants: Pour chaque cause de défaillance, nous calculons un indicateur de risque (RPN) correspondant, défini comme le produit de ces trois paramètres:

$$\text{RPN} = S * O * D \quad (1)$$

Le RPN attribue un poids à chaque mode de défaillance considéré. Plus la valeur du RPN est élevée plus le produit est critique. Pour un composant donné, l'agrégation des RPN émanant de chaque cause racine donne ensuite le RPN du mode de défaillance sélectionné. L'agrégation des RPN émanant de chaque mode de défaillance du composant construit le RPN total du composant. Pour calculer le RPN du sous-système, chaque effet du mode de défaillance du composant correspond au mode de défaillance du sous-système. L'agrégation des RPN des composants pour chaque mode de défaillance génère le RPN total du sous-système. Ainsi, nous comparons la criticité de chaque sous-système à partir de l'espace de conception et nous éliminons les plus critiques. Cette phase traduit le 3^{ème} principe du SBCE « développement d'un ensemble de concept ».

Après cette phase, les spécialistes de sous-systèmes renvoient un flux ascendant contenant le feedback sur leur analyse spécifique dans le compartiment de criticité du bloc des sous-systèmes, comme illustré dans la figure 3.2. Un premier filtre est effectué par comparaison de la criticité des sous-systèmes. Le sous-système le plus critique est désactivé de l'ensemble des différentes alternatives du diagramme de définition de blocs et de la matrice d'allocation.

La quatrième phase de cette approche correspond au quatrième principe du SBCE « convergence des concepts ». Après la phase d'intégration, une analyse de robustesse de chaque système conçu doit être effectuée pour converger vers un ensemble de solutions optimisées. Pour étudier la robustesse des systèmes conçus, quelques domaines spécifiques telles que la fiabilité, la durabilité et la performance de chaque système sont considérées pour converger vers un nombre limité de solutions optimales. Dans notre approche, un algorithme d'analyse de la fiabilité est réalisé pour effectuer une analyse de domaine spécifique dans un environnement SysML et pour réduire l'espace de conception. L'analyse de la durabilité et de la performance serviront pour l'étude de compromis.

Dans cette phase, les parties prenantes ont la possibilité de travailler de deux manières:

- Premièrement, séquencer la prise de décision pour adapter les phases de conception aux informations existantes en raison de l'absence de certaines valeurs de KVA.
- Deuxièmement, la méthode simultanée lorsque toutes les informations sur le KVA sont utilisées (toutes les valeurs de KVA sont disponibles).

Pour étudier la fiabilité des systèmes conçus, l'outil utilisé est l'analyse par arbre de défaillances, qui est une technique d'analyse des systèmes utilisée pour déterminer la probabilité d'occurrence d'un événement indésirable prédéfini [90]. Les données d'entrée sont extraites des diagrammes SysML définis conformément au diagramme de profil SysML, qui correspondent aux

différentes probabilités de défaillance des composants et au type de la porte entre les blocs.

L'arbre de défaillance contient trois niveaux hiérarchiques:

- Le premier niveau représente l'événement principal, qui traduit la défaillance du système.
- Pour le deuxième niveau, chaque bloc représente la cause de défaillance avec une probabilité d'affecter l'événement indésirable. Cette cause pourrait être une défaillance électrique, mécanique ou électronique.
- Le troisième niveau représente l'événement de base, qui traduit la défaillance d'un composant.

Dans le premier cas, si le type de flux entre le bloc de sous-système et le bloc de composant ou entre le bloc d'activité principal et le bloc de sous-système est la porte OU, alors la probabilité de défaillance P du bloc n est présentée dans l'équation 2:

$$P(n) = \sum p_i(n) - \prod P_i(n) \quad (2)$$

Dans cette équation, i représente tous les blocs successeurs de n . Dans le second cas, si le type de flux entre le bloc de sous-système et le bloc de composant ou entre le bloc d'activité principal et le bloc de sous-système est la porte ET, alors la probabilité de défaillance P du bloc n est présentée dans l'équation 3:

$$P(n) = \prod P_i(n) \quad (3)$$

Chaque spécialiste effectue des analyses de domaine spécifique pour étudier la robustesse des systèmes conçus et élimine les solutions systèmes qui ne sont pas faisables ou qui sont coûteuses. Par conséquent chaque spécialiste génère un ensemble de solutions faisables. Les solutions générées à partir de chaque analyse sont ensuite implémentées dans le modèle MBRA à travers le diagramme de profil. Dans ce diagramme, l'activité principale est stéréotypée. Elle

comporte la fiabilité, le coût, la durabilité et la performance d'un ensemble de solutions systèmes réduits. L'intersection des solutions indépendantes générées à partir de ces analyses définira l'ensemble des solutions final, comme indiqué dans la figure 3.1.

La dernière étape consiste à adopter une analyse de compromis effectuée par l'ingénieur système en se basant sur le feedback des spécialistes. Le concept de compromis implique un choix tactique ou stratégique avec la compréhension des avantages et des inconvénients de chaque configuration. Il vise également à choisir la meilleure architecture système répondant au mieux aux besoins des clients. Au cours de cette phase de conception, l'ingénieur système recueille les feedbacks provenant des départements de la fiabilité, de la durabilité, de la performance et du calcul du coût pour choisir la meilleure solution système parmi l'ensemble réduit de solutions optimisées. L'objectif de cette phase repose sur l'utilisation l'équation suivante:

$$F(n) = \sum_{i=1}^n \alpha(i) T(i) \quad (4)$$

Où n est le nombre de compromis, $\alpha(i)$ est le coefficient de pondération extrait de l'analyse AHP et $T(i)$ est la valeur de chaque compromis.

3.3. Implémentation de l'algorithme

En raison de la difficulté d'implémenter notre approche dans un outil basé sur SysML, nous avons développé une preuve de concept en utilisant le langage Python pour effectuer l'analyse de fiabilité. La première étape consiste à extraire les attributs des composants des diagrammes SysML à travers un fichier XMI. Les valeurs extraites sont ensuite transformées en graphe à l'aide d'un outil appelé GEPHI [91].

Le graphe initial est un ensemble de nœuds et d'arcs entre des paires de ces nœuds. L'utilisation d'un graphe présente l'avantage de visualiser une relation parmi un ensemble, ce qui facilite la

compréhension des caractéristiques des relations. Les solutions systèmes et leurs sous-systèmes et composants respectifs sont représentés avec des nœuds. Chaque nœud contient des attributs tels que la probabilité de défaillance et le type de portes. Les liens entre les nœuds représentés par des arcs traduisent les facteurs d'influence entre les sous-systèmes. A partir du graphe initial généré par Gephi, un fichier GraphML est exporté ensuite vers le langage Python, pour lire le graphe initial et pour filtrer les solutions faibles pour l'analyse de la fiabilité. L'organigramme de cette analyse est présenté dans la figure 3.3.

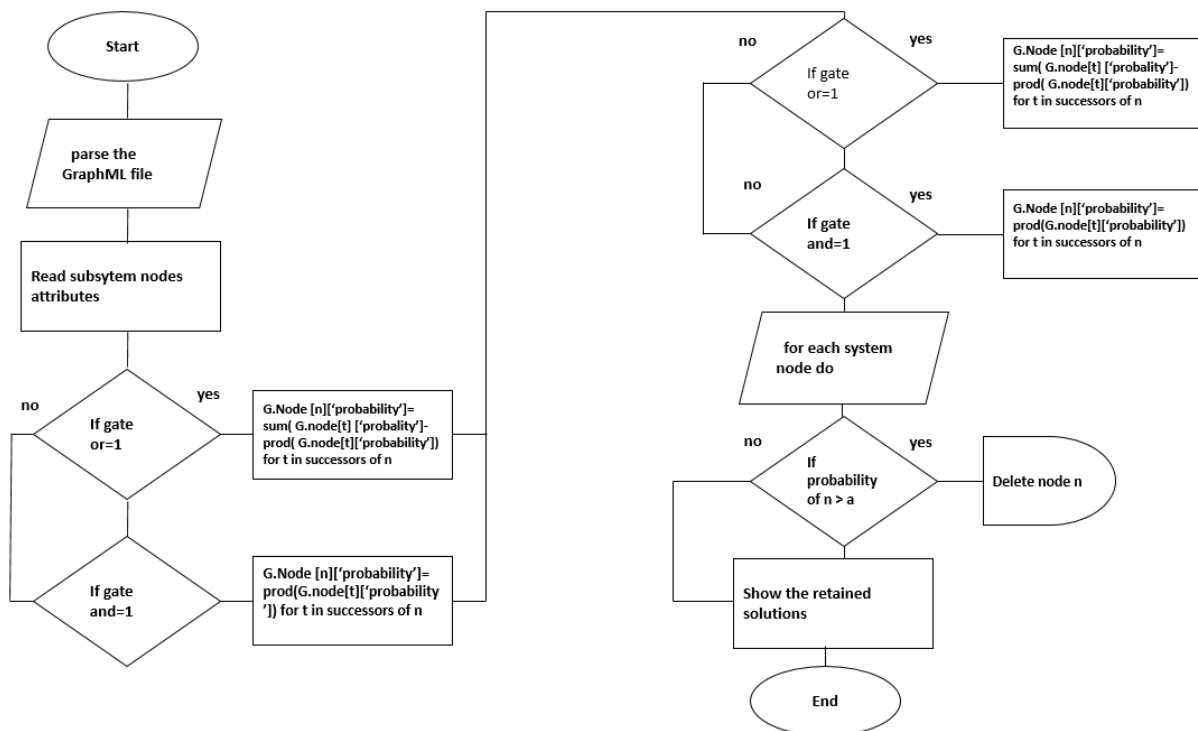


Fig.3.3 :l'organigramme de l'analyse de la fiabilité

L'application de l'algorithme développé pour l'analyse de fiabilité permet de converger vers un ensemble de solutions réduit.

Les solutions alternatives restantes de l'espace de conception sont évaluées à l'aide d'un algorithme de compromis. En effet, l'entrée de cet algorithme est constituée des différents attributs du système, tels que la fiabilité, le coût, la durabilité et la performance classés de 1 à

5, afin d'aider l'ingénieur système à choisir la meilleure architecture. 1 correspond à la valeur la plus défavorable, tandis que 5 correspond à la meilleure donnée par l'expert. Le résultat de cet algorithme est illustré dans un diagramme radar.

3.4. Etude de cas d'un compresseur électrique

L'industrie automobile est en train d'innover de nouvelles fonctionnalités qui améliorent le confort et la sécurité des passagers, l'économie du carburant des véhicules et qui réduisent les émissions de CO₂ dans l'environnement. Les compresseurs électriques, qui font partie du système de climatisation, présentent les avantages suivants: faibles émissions de réfrigérant, conditionnement flexible et fonctionnement efficace à vitesse variable [92]. Ce compresseur aspire un réfrigérant à basse pression à travers l'évaporateur. Ensuite, il comprime le réfrigérant et le transmet sous forme de gaz chaud au condenseur (côté haute pression du circuit de réfrigérant). Cette pièce automobile constitue ainsi l'interface entre les côtés basse et haute pression du circuit frigorifique. Pour le développement des compresseurs électriques, l'architecture des compresseurs conventionnels a été considérablement modifiée dans le but de réduire les dimensions et le poids tout en maintenant la même capacité de réfrigérant.

La figure 3.4 illustre le compresseur électrique contenant une partie commande, un transformateur, un moteur et une partie mécanique.

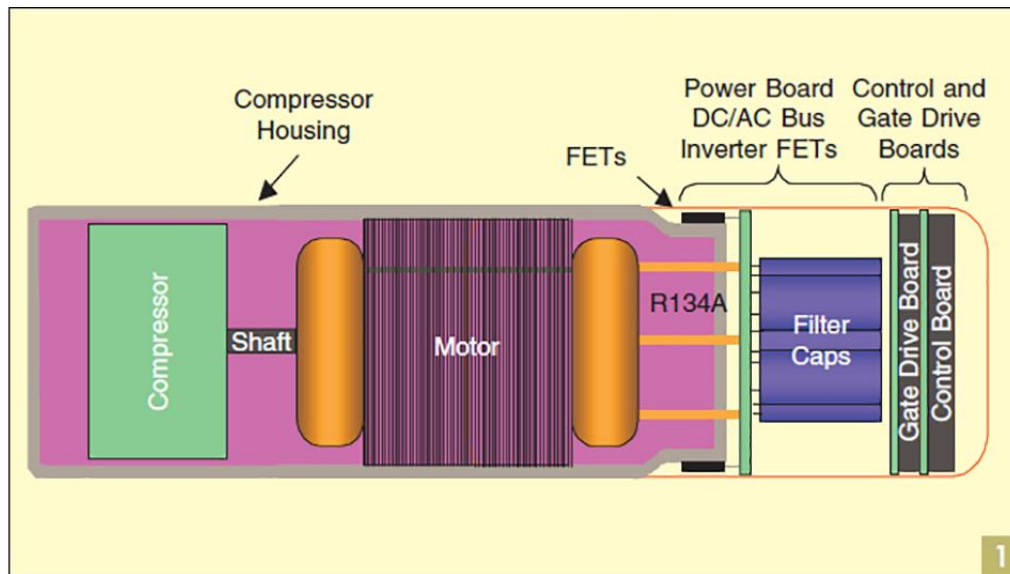


Fig.3.4 : le compresseur électrique [92]

L'approche utilisée pour développer ce système mécatronique est décrite comme suit:

La première étape consiste à définir les valeurs du client. Tout d'abord, les spécifications des clients permettent aux parties prenantes d'identifier 1000 valeurs.

Ensuite, ces valeurs sont classées en valeurs singulières, qui sont la fiabilité, le coût, la performance et la durabilité. Enfin, les parties prenantes les classent des plus importantes aux moins importantes à l'aide d'outils basés sur l'AHP, comme indiqué dans le tableau 2:

Attributs de valeurs clé	Nombre de valeurs	Critère d'importance
Fiabilité	374	$374/1000 = 37.4\%$
Coût	299	$299/1000=29.9 \%$
Performance	197	$197/1000=19.7\%$
Durabilité	130	$130/1000=13\%$

Tab.2 : les attributs de valeurs clé

Les exigences de niveau supérieur sont décomposées en exigences plus détaillées. Ensuite, elles sont introduites dans le diagramme d'exigences de SysML comme illustré à la figure 3.5.

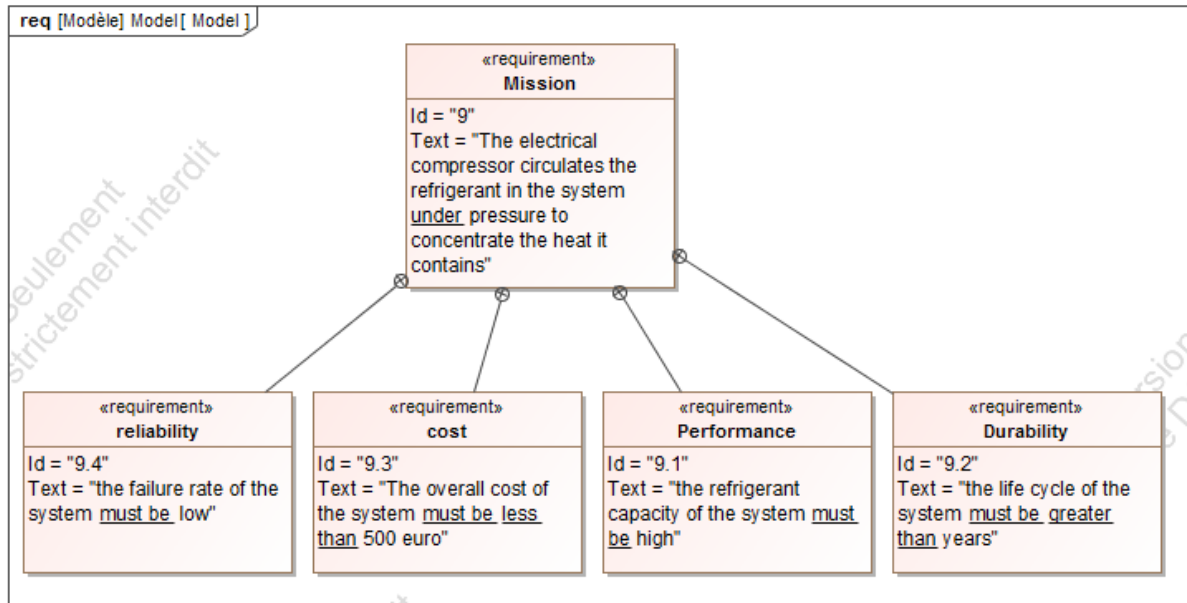


Fig.3.5 : Le diagramme d'exigences SysML du compresseur électrique

La troisième étape consiste à définir l'architecture fonctionnelle du compresseur électrique. Pour clarifier la fonctionnalité du système, une itération entre les exigences détaillées et les fonctions dérivées est répétée jusqu'à identifier les fonctions ciblées. Les exigences de niveau supérieur sont décomposées en exigences détaillées. De même, les fonctions peuvent être décomposées en sous-fonctions. Cette activité est appelée la décomposition fonctionnelle et son résultat est l'architecture fonctionnelle du système. Par conséquent, dans ce modèle, il existe trois niveaux hiérarchiques des fonctions :

- La fonction du système
- La fonction du sous-système
- la fonction du composant.

La figure 3.6 illustre le diagramme d'activité.

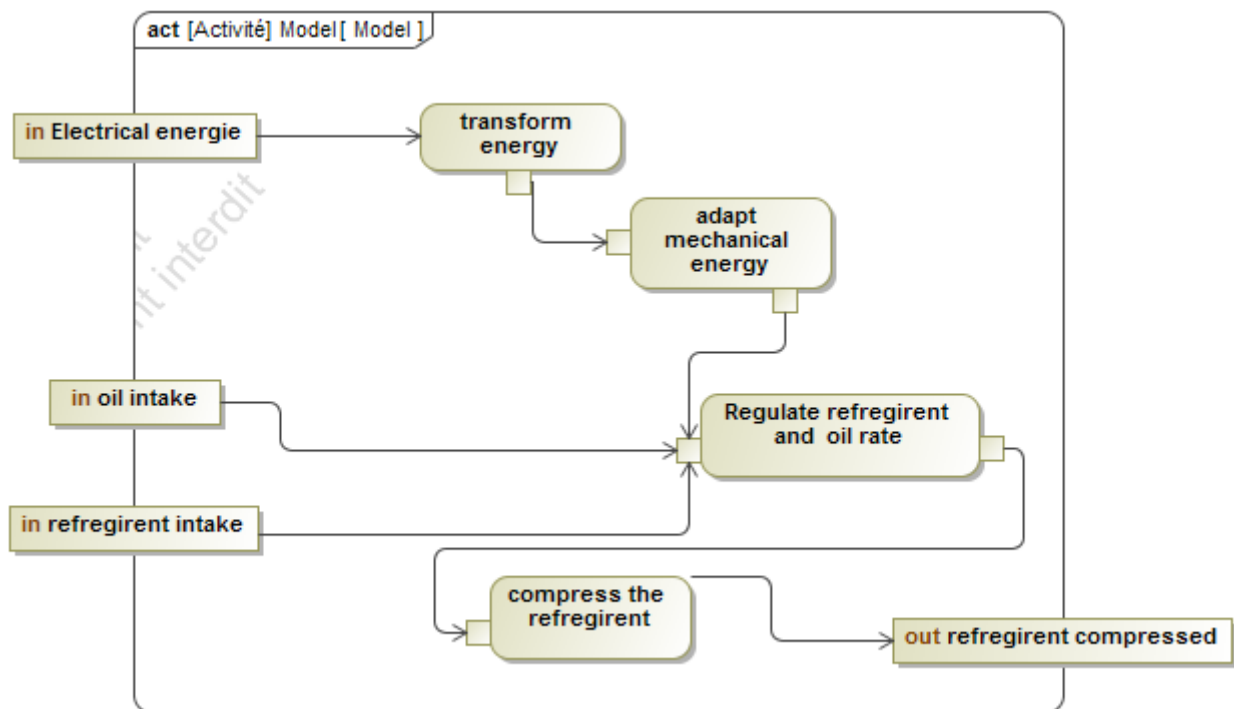


Fig.3.6 : Le diagramme d'activités SysML du compresseur électrique

Dans ce diagramme, l'activité principale du compresseur électrique est décomposée en sous-fonctions. En effet pour comprimer l'air, nous avons besoin d'une unité qui transforme l'énergie électrique en énergie mécanique et une autre unité qui adapte cette énergie mécanique. Ceci construit l'architecture fonctionnelle du système et permet donc au concepteurs de mieux comprendre le fonctionnement de ce système. L'étape suivante consiste à définir l'architecture d'allocation, comme illustré à la figure 3.7.

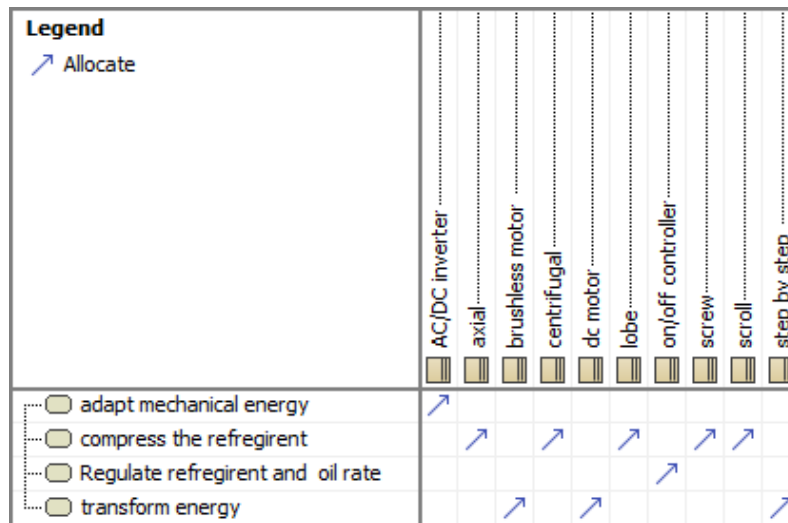


Fig.3.7 : Matrice d'allocation SysML du compresseur électrique

Dans ce diagramme, chaque activité est allouée à une ou plusieurs solutions techniques de sous-système. Par exemple, pour transformer l'énergie, les solutions techniques identifiées sont : moteur à courant continu, moteur sans balais ou moteur pas à pas. Cette architecture permet de représenter le taux d'innovation de l'entreprise et la portée du travail des ingénieurs.

La quatrième étape consiste à développer l'architecture physique en se basant sur la matrice d'allocation. Cette architecture est implémentée dans le modèle MBRA à l'aide du diagramme de définition de blocs (BDD). Les objectifs de chaque sous-système sont bien définis afin de prévenir un travail excessif des ingénieurs et de soutenir le développement de l'innovation. Cela définira la portée du travail à partir de laquelle les concepteurs et les ingénieurs pourront explorer et communiquer les solutions alternatives de sous-systèmes et de composants avec d'autres parties prenantes. Nous supposons que l'architecture physique contient 3 niveaux hiérarchiques (système, sous-systèmes et composants) pour faciliter l'analyse de domaine spécifique à partir des diagrammes SysML et pour éviter le problème d'explosion combinatoire lors de la phase d'intégration. La figure 3.8 illustre l'architecture physique du système avec l'intégration des nouveaux stéréotypes de flux développés.

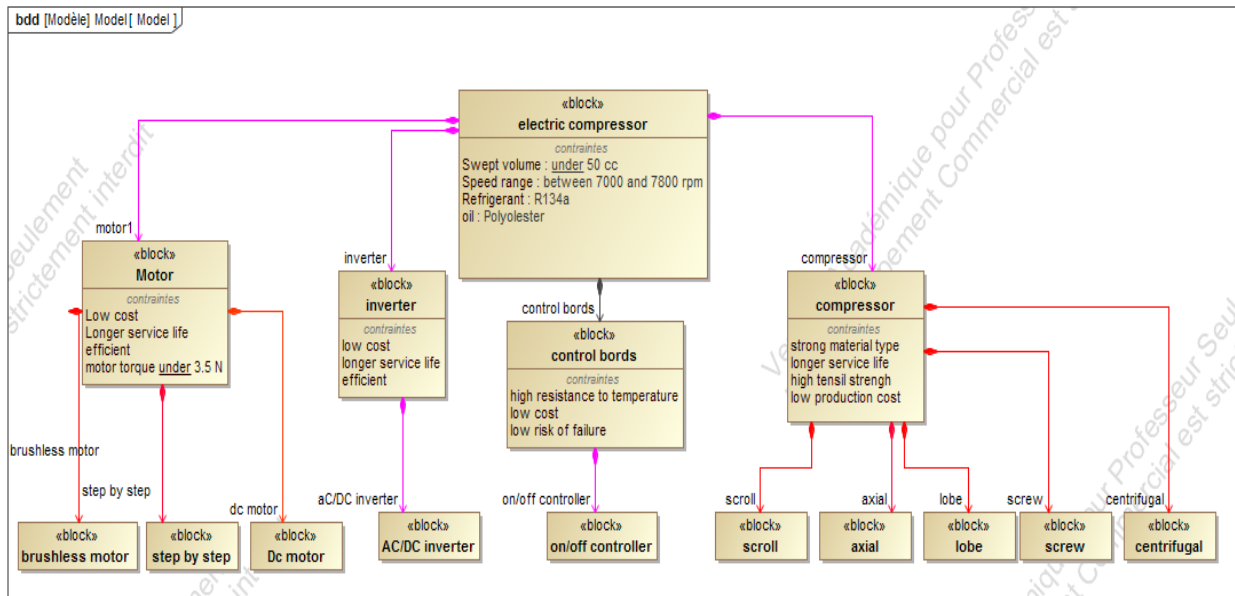


Fig.3.8 : L'architecture physique du compresseur électrique intégrée dans le diagramme de définition de blocs SysML

L'architecture physique du système est constituée de 3 niveaux : l'activité principale est représentée par le bloc « compresseur électrique ». Ce bloc contient quelques contraintes fonctionnelles tels que le volume qui ne doit pas dépasser 50cc, la vitesse qui doit être entre 7000 et 7800 tr/min, le type du réfrigérant (R134a) et le type de l'huile (polyolester). Le 2^{ème} niveau contient la partie commande, le transformateur, le moteur et la partie mécanique du compresseur. Les blocs des sous-systèmes « moteurs » et « compresseurs » contiennent une porte « ou » donc les flux de composition sont présentés en rouge. Pour les blocs « transformateur » et « unité de contrôle », une seule solution technique est identifiée dans l'espace de conception donc le flux de composition est présenté en violet. L'intégration des sous-systèmes pourrait générer 15 solutions systèmes potentiels dans l'espace de conception (3 moteurs* 1 transformateur* 1 carte commande * 5 parties mécaniques du compresseur).

Dans une étape plus avancée de conception, la définition des solutions techniques de composants permet d'effectuer des analyses de domaines spécifiques et de filtrer les solutions faibles ou coûteuses. Par exemple, pour les moteurs, les experts définissent les échelles de

détection, d'occurrence et de gravité sur la base de la norme MIL-STD-1629A. Le tableau 3 présente les échelles d'évaluation de la gravité des moteurs.

Echelle	Description	Critère
1	marginale	L'électricité peut être générée
2	mineur	Réduction de la capacité de générer de l'électricité
3	modéré	Dégradation progressive des performances
4	élevé	Perte de capacité à produire de l'électricité
5	catastrophique	Dommages majeurs au moteur

Tab.3: Les critères d'évaluation de la gravité des moteurs

Les critères de détection et de l'occurrence sont définis de la même manière. L'étape suivante consiste à définir les modes de défaillance et les causes racines correspondantes. Pour les moteurs, les modes de défaillance détectés sont indiqués dans le tableau 4:

Mécanique	Electrique	Matériel
thermique	Panne électrique	Fatigue
rupture	Isolation électrique	construction

Tab.4 : Les modes de défaillance des moteurs

Pour chaque mode de défaillance, les causes racines correspondantes sont identifiées dans le tableau 5 :

Mécanique	Electrique	Matériel
Surcharge mécanique	Court circuit	Fatigue de vibration
Défaut de fabrication	Surcharge électrique	surchauffe
Dommages externes	Perte de puissance	corrosion

Tab.5 : Les causes racines de chaque mode de défaillance des moteurs

Pour bien appliquer l'outil AMDEC, il est essentiel de prendre en compte le niveau de détail nécessaire pour une représentation fidèle du système sans compliquer l'analyse. Si le système est décomposé en composants individuels, il devient complexe et requiert une connaissance approfondie du système. Pour les compresseurs électriques, les détails sur tous les composants ne sont pas disponibles dans le catalogue fournisseur, donc il est acceptable d'appliquer l'outil

AMDEC jusqu'au niveau des sous-ensembles. Pour le cas du moteur sans balais, les attributs des composants sont définis à l'aide du diagramme BDD comme le montre la figure 3.9 :

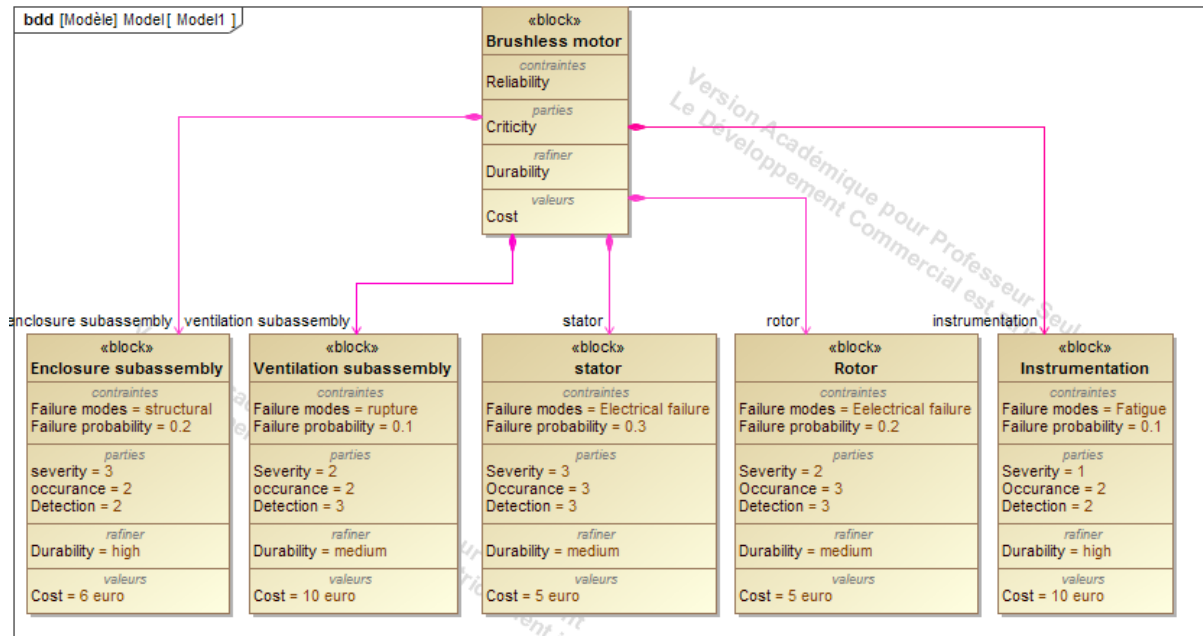


Fig. 3.9 : Attributs des composants du moteur sans balais intégrés dans le diagramme de définition de blocs SysML

Ce diagramme définit les attributs des composants du sous-système « Moteur sans balais ». Pour chaque mode de défaillance détecté, les experts évaluent les critères de sévérité, d'occurrence et de détection de 1 à 4.

La phase de test consiste à comparer les criticités des sous-systèmes et d'éliminer les solutions dont la criticité dépasse une certaine valeur définie par les parties prenantes. En effet, chaque criticité de sous-système avec un $RPN > 100$ est filtrée de l'espace de conception. Ce critère d'élimination a été défini à travers plusieurs sessions de brainstorming au sein d'une équipe de recherche d'experts en fiabilité et en qualité. Après la phase de test, le moteur pas à pas et la partie mécanique du compresseur centrifuge dépasse le seuil défini donc ces deux sous-systèmes sont éliminés. Ainsi, l'espace de conception est réduit à 8 solutions potentielles: 2

moteurs * 4 compresseurs * 1 transformateur * 1 carte de contrôle. Cette étape traduit le 3^{ème} principe de l'approche SBCE.

Pour analyser la robustesse des systèmes, un deuxième filtre est appliqué après la phase d'intégration pour réduire le nombre de solutions système et pour converger vers un nombre de solutions optimales. Dans cette étape, chaque spécialiste effectue une analyse de domaine spécifique à partir des données extraites du diagramme de profil SysML. Dans ce cas d'étude, les domaines de durabilité et de performance ne seront pas détaillés. Seulement l'analyse de la fiabilité sera appliquée afin de sélectionner les solutions les plus performantes.

Premièrement, les attributs sont introduits dans un tableau de Gephi. Ces attributs sont porte « ET », porte « OU » et la probabilité de défaillance. Ce tableau est ensuite traduit en graphe sous forme d'arbre de défaillance comme le montre la figure 3.10 :

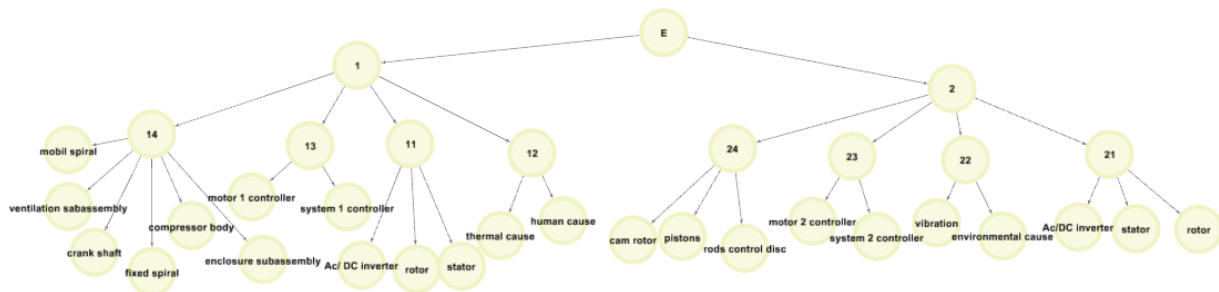


Fig.3.10 : le graphe initial généré par Gephi pour l'étude de la fiabilité

Ce graphe contient quelques solutions systèmes de l'espace de conception. Chaque solution contient 3 niveaux hiérarchiques. Le premier niveau contient les nœuds libellés de [1..n]. Ces nœuds représentent l'évènement indésirable qui est la défaillance du système. Le deuxième niveau contient les nœuds libellés [1₁, 1₂,..., n₁, n₂..]. Ces nœuds représentent les causes de défaillance du système. Ces causes peuvent provenir de la partie électrique, de la partie commande, de la partie mécanique ou d'une cause extérieure. Le dernier niveau contient les nœuds qui représentent les composants de chaque système conçu. Chaque nœud de composant

contient une probabilité de défaillance. La deuxième étape consiste à analyser ces attributs avec Python. Ensuite, une itération est effectuée pour calculer la probabilité de défaillance du sous-système en fonction des types de portes. Pour la dernière étape de cet algorithme, chaque probabilité de défaillance des systèmes générés est calculée en tenant compte du mode de défaillance du sous-système.

Après l'application de la programmation, l'espace de conception est réduit à 3 solutions. Un flux ascendant contenant le résultat de cette analyse est ensuite intégré dans le diagramme de profil SysML afin d'aider l'ingénieur système à mener une étude de compromis. Pour choisir la meilleure architecture système, les résultats de chaque domaine spécifique pour l'ensemble final des solutions systèmes sont présentés dans un diagramme radar. La position relative et l'angle des axes sont généralement non informatifs, comme illustré à la figure 3.11.

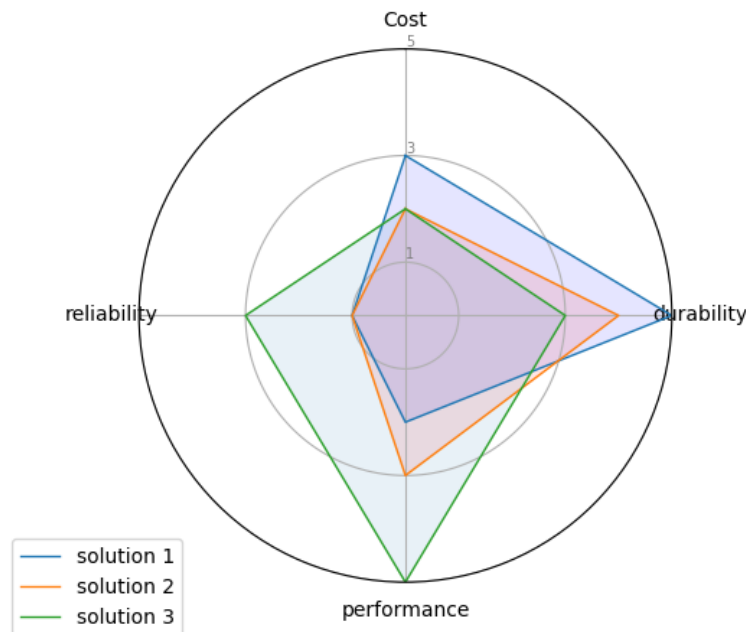


Fig.3.11 : Le diagramme radar généré par Python pour l'analyse des compromis

La meilleure solution est celle qui optimise tous les compromis, c'est à dire la fiabilité, la performance, le coût et la durabilité. Après cette analyse des compromis, la solution finale est communiquée à d'autres parties prenantes pour commencer la conception détaillée.

3.5. Comparaison avec des travaux similaires

De nombreux travaux de recherche dans la littérature utilisent des approches basées sur des ensembles et des approches basées sur les connaissances afin d'optimiser les délais et de minimiser les boucles itératives lors de développement des systèmes complexes comme le travail effectué par [93]. Dans ce travail de recherche, une approche descendante est utilisée pour effectuer une décomposition fonctionnelle et comportementale du système afin de définir des cibles de sous-système à travers l'utilisation d'une taxonomie d'un concept intitulé «pièce intelligente» qui transforme les spécifications du système en une architecture physique. Ceci est similaire à notre travail. En effet, l'adoption d'une approche descendante «top down » permet de bien définir les exigences fonctionnelles et de converger vers les solutions techniques qui répondent aux spécifications des clients. En outre, cette taxonomie permet d'assurer la traçabilité des contraintes de conception. Néanmoins, ce concept de pièce intelligente ne permet pas de fournir un environnement d'ingénierie simultanée où les équipes pluridisciplinaires communiquent leurs idées et leurs analyses de domaines spécifiques. Ceci peut générer des boucles itératives dans la phase de conception détaillée.

Des méthodologies similaires sont utilisées pour éviter les boucles itératives entre les ingénieurs comme celle développée par [94]. En effet, cette méthodologie est basée sur une approche modulaire et centrée sur les connaissances des experts pour bien définir l'architecture physique du système. Cette méthodologie présente des similitudes avec le modèle MBRA. En premier lieu, l'objectif de ce travail est de minimiser le temps de développement en éliminant les tâches sans valeur ajoutée et en automatisant les tâches répétitives. En deuxième lieu, la collaboration des parties prenantes lors des phases de conception préliminaires permettra aux ingénieurs débutants d'acquérir de l'expérience auprès d'experts en communiquant les

différentes solutions de composants et de sous-systèmes avec d'autres équipes d'un même environnement.

Dans le même contexte, une approche développée dans [95] est utilisée pour optimiser l'utilisation du temps et des ressources. Elle vise à réduire la complexité des tâches en décomposant les exigences et les contraintes de niveau supérieur en contraintes locales afin que chaque composant ou sous-système soit traité séparément. Ce travail est similaire au modèle MBRA. En effet, dans notre modèle les équipes de sous-systèmes définissent la portée du travail en se basant sur le taux d'innovation de l'entreprise. Ceci évitera la surcharge du travail des concepteurs. Ensuite, ces équipes partagent leurs analyse dans le modèle MBRA avec d'autres parties prenantes pour bien définir les attributs des composants dans une étape plus avancée de conception. Ceci permettra d'éviter les tâches inutiles et donc de minimiser le temps de développement.

3.6.Discussion

L'étude de cas du compresseur électrique donne un aperçu sur l'application d'ingénierie simultanée basée sur les ensembles (SBCE). Premièrement, à travers la compréhension des spécifications du client, les parties prenantes peuvent définir la portée du travail requise en transformant ces spécifications en attributs de valeurs clé. Grâce au processus d'hierarchie analytique (AHP), les valeurs détectées sont la fiabilité, le coût, la performance et la durabilité. Ces attributs de valeurs clé ont pour objectif de définir le taux d'innovation requis pour chaque sous-système et de s'aligner avec la stratégie de l'entreprise.

Deuxièmement, Pour cartographier l'espace de conception, une itération entre les exigences détaillées et les fonctions dérivées est établie pour assurer une bonne compréhension du fonctionnement du système. Ensuite, l'utilisation de l'architecture d'allocation permet d'allouer une fonction à une ou plusieurs solutions techniques. Les ingénieurs de sous-systèmes peuvent

ainsi partager leurs idées et travailler librement au début du processus de conception afin d'éviter le gel des solutions. Pour le compresseur électrique, l'espace de conception de sous-systèmes pourrait générer 15 solutions systèmes potentielles après la phase d'intégration : (3 solutions pour le moteur* 1 solution pour le transformateur * 1 solution pour l'unité de contrôle* 5 solutions de la partie mécanique du compresseur). L'avantage considérable de cette étape est d'éviter les boucles itératives entre les ingénieurs dans la phase aval de conception en imposant un nombre minimal de contraintes.

Afin d'implémenter cet ensemble de solutions dans les diagrammes de SysML, nous avons développé un diagramme de profil qui contient de nouveaux stéréotypes de flux et de nouveaux stéréotypes de blocs. Ce diagramme définit les attributs liés aux systèmes, aux sous-systèmes et aux composants. Il permet ainsi aux équipes pluridisciplinaires de faire des analyses de domaines spécifiques et d'assurer la traçabilité de chaque analyse.

Dans une étape plus avancée de conception, la définition des solutions techniques des composants permet aux participants à la conception de maîtriser la complexité de développement de ce système mécatronique et de converger plus rapidement vers un ensemble de solutions systèmes réduit.

Pour tester les sous-systèmes, l'utilisation de l'outil AMDEC est un moyen efficace pour éliminer les solutions les plus critiques. En effet, pour assurer cette phase de test, les attributs utilisés sont le mode défaillance, la valeur de la sévérité, la valeur d'occurrence et la valeur de détections pour chaque mode. Ces attributs permettent ainsi de calculer la criticité de chaque sous-système. Dans cette phase, nous supposons que chaque mode de défaillance a un seul effet et une seule pour faciliter la phase de test. Vu le manque de détails sur tous les composants, nous avons appliqué l'outil AMDEC jusqu'au niveau des sous-ensembles. Après cette phase de

test, l'espace de solutions est réduit à 8 solutions systèmes potentielles (2 solutions pour le moteurs* 1 solution pour le transformateur * 1 solution pour l'unité de contrôle* 4 solutions pour la partie mécanique du compresseur).

Pour appliquer le 4^{ème} principe de l'approche SBCE « convergence des concepts », nous avons développé un algorithme pour l'analyse de la fiabilité des systèmes conçus en se basant sur l'outil arbre de défaillance. Cet algorithme permet d'analyser simultanément l'ensemble de solutions afin de converger vers un ensemble de solutions fiables. Pour implémenter cet algorithme, nous avons utilisé un proof de concept qui se base sur l'outil Géphi et sur le langage Python pour éliminer les solutions qui présentent un grand risque de défaillance. Cette étape permet ainsi de générer un gain de temps considérable par rapport à l'approche traditionnelle par point. Ce proof de concept pourrait être implémenté par les éditeurs de logiciels qui utilisent SysML comme sous forme de Plugins pour assurer une analyse de domaine spécifique.

La dernière étape consiste à évaluer l'ensemble final de solution à travers une étude de compromis. Pour cela, nous avons développé un algorithme qui permet de transformer les valeurs des compromis (la fiabilité, le coût, la performance et la durabilité) de chaque solution système en un diagramme radar. L'avantage de cette étape est d'assister l'ingénieur système lors du choix de la meilleure architecture système.

Pour conclure, ce nouveau modèle MBRA contribue tout d'abord à améliorer l'efficacité de développement des produits complexes tels que les systèmes mécatroniques en offrant aux équipes pluridisciplinaires une liberté de décision au début de la phase de conception. Ceci permettra d'éviter des reconceptions coûteuses lors des phases de conception détaillées. Un autre avantage de ce modèle est de construire un capital de connaissance par la collaboration avec les experts en fiabilité et en qualité. Ainsi les ingénieurs inexpérimenté peuvent concevoir

des syst mes robustes et plus fiables. Enfin, en int grant les principes de l'     dans le processus de d veloppement des syst mes complexes, les participants   la conception peuvent converger plus rapidement vers la solution qui r pond au mieux aux exigences des clients par rapport   l'approche traditionnelle par point.

Chapitre 4

Développement du modèle MBMA pour l'étude de la fabricabilité

4.1.	Introduction.....	87
4.2.	L'approche proposée.....	88
4.3.	Implémentation de l'algorithme	97
4.4.	Etude de cas d'un boîtier papillon.....	99
4.5.	Discussion	114

4.1. Introduction

Dans ce chapitre, une approche systématique est utilisée pour étudier la fabricabilité d'un ensemble de procédés de fabrication avant la phase d'industrialisation en intégrant l'ingénierie systèmes basée sur les modèles MBSE avec les principes du SBCE. L'objectif de cette approche est de fournir un cadre de développement au sein duquel un ensemble de parties prenantes travaille simultanément afin de converger rapidement vers la meilleure solution de processus de fabrication parmi un ensemble, qui optimise le coût, le temps et la qualité de la pièce. Un autre objectif de cette approche est d'intégrer les contraintes métiers telle que la fabrication dès la phase de conception. Selon [96], «une erreur de fabrication découverte et corrigée au cours de la phase de conception est relativement peu coûteuse à corriger. Si cette même erreur est découverte lors de la phase de fabrication, la correction peut être jusqu'à 100 fois plus coûteuse en raison des investissements déjà réalisés et perdus, du temps de préparation pour la fabrication et la reconception, ainsi que le temps perdu avant la mise sur le marché ». L'approche proposée repose sur le développement d'un nouveau modèle intitulé « Model Based for Manufacturability Analysis » (MBMA). Ce modèle est basé sur deux types de flux d'informations: Un flux descendant qui représente l'échange d'informations émises par l'ingénieur système vers les différents spécialistes, tandis que le flux ascendant sert de feedback pour aider l'ingénieur système à choisir le meilleur processus de fabrication en se basant sur les ressources disponibles dans l'atelier de fabrication, les contraintes de la pièce et les connaissances des experts. Pour cela, les principes du SBCE sont convertis à un cadre adapté à la fabrication. En effet, l'espace de conception est remplacé par un ensemble de processus de fabrication. SysML est utilisé comme outil descriptif pour assurer la traçabilité des contraintes et pour communiquer les différentes alternatives entre les parties prenantes. La contribution principale de ce nouveau modèle est sa capacité d'évaluer simultanément un groupe de

processus de fabrication afin de diminuer les boucles itératives entre les participants à la conception en adoptant des filtres. Une autre contribution de ce modèle est la quantité d'informations explorées à chaque niveau de développement des processus. En fait, cela pourrait constituer un capital de connaissances pour les concepteurs afin qu'ils puissent améliorer leur conception et converger vers la meilleure solution plus rapidement. Dans la section suivante, une approche systématique est développée pour intégrer les contraintes métiers dans les diagrammes SysML afin d'explorer l'ensemble des procédés de fabrication. La troisième partie « Implémentation de l'algorithme » est dédiée à l'application d'un algorithme qui permet de converger vers la meilleure solution. Enfin, dans la 4^{ème} section, une étude de cas d'un corps de papillon électronique est présentée pour illustrer les différentes étapes de cette approche d'ingénierie simultanée.

4.2. L'approche proposée

L'approche proposée a pour objectif de fournir un modèle commun « MBMA » d'analyse de la fabricabilité d'un système mécatronique. Ce modèle sera utilisé simultanément par les différentes parties prenantes qui sont les ingénieurs métiers (ingénieur de fabrication, concepteurs, ingénieur de production) et les experts. Dans ce modèle, les principes de l'approche SBCE sont convertis à un cadre adapté à la fabrication pour cartographier l'espace de solutions et pour effectuer une comparaison simultanée d'un ensemble de processus de fabrication. En outre, cette approche permet de développer des connaissances sur plusieurs concepts de fabrication en ajoutant des informations issues des tests et de l'analyse de la fabricabilité. Par conséquent, les solutions faibles ou coûteuses sont filtrées successivement de l'espace de solutions jusqu'à la convergence vers la meilleure solution. Un procédé est conservé dans l'espace de solution s'il satisfait toutes les contraintes de conception. La figure 4.1 illustre le modèle MBMA développé :

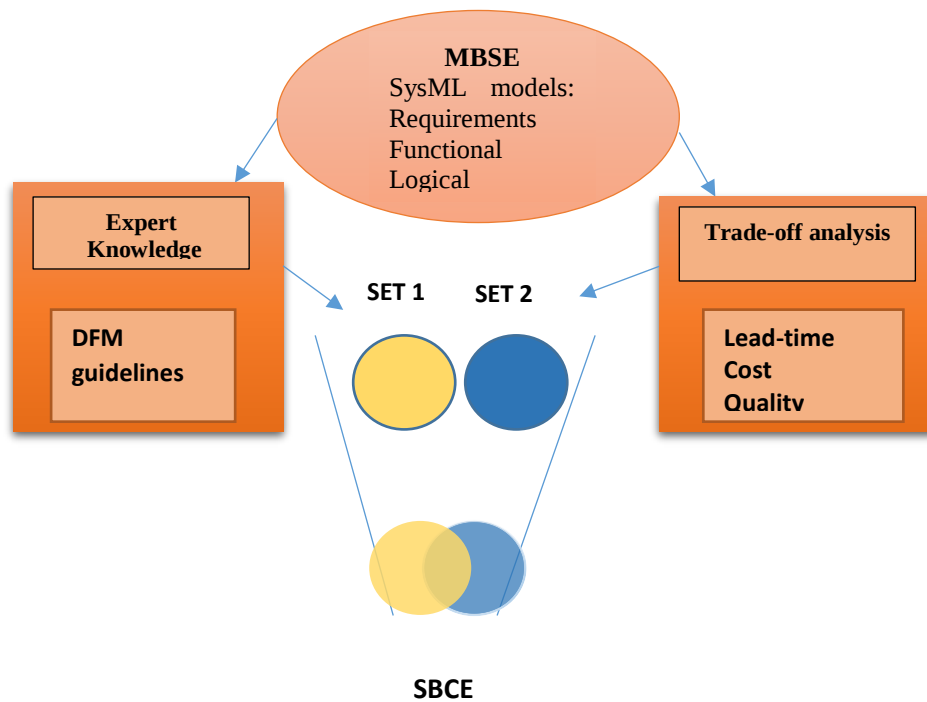


Fig.4.1 : Le modèle MBMA

À travers ce modèle descendant, trois aspects du système sont pris en compte (les exigences, l'aspect fonctionnel et l'aspect logique) et illustrés dans des diagrammes SysML. Ce modèle commence par examiner les spécifications du client avant de converger vers un nombre limité de solutions réalisables. L'intersection de ces ensembles donnée par les différentes parties prenantes conduit à la meilleure solution. Le modèle MBMA est structuré comme suit:

- Une phase descriptive contenant des diagrammes SysML pour définir les exigences et les contraintes liées aux spécifications du client, à la conception de la pièce, aux processus de fabrication et aux ressources de l'atelier de fabrication. Dans cette phase descriptive, un diagramme de profil (figure 4.2) est développé pour définir des stéréotypes de bloc lié aux

entités de fabrication de la pièce et aux processus de fabrication. Au fur et à mesure que l'analyse progresse, les parties prenantes peuvent ajouter ou supprimer des informations nécessaires pour définir la relation entre les contraintes de la pièce et chaque processus de fabrication.

- Une phase de calcul contenant les données extraites des diagrammes de SysML développés à partir du diagramme de profil pour effectuer une analyse de compromis dans l'environnement SysML.

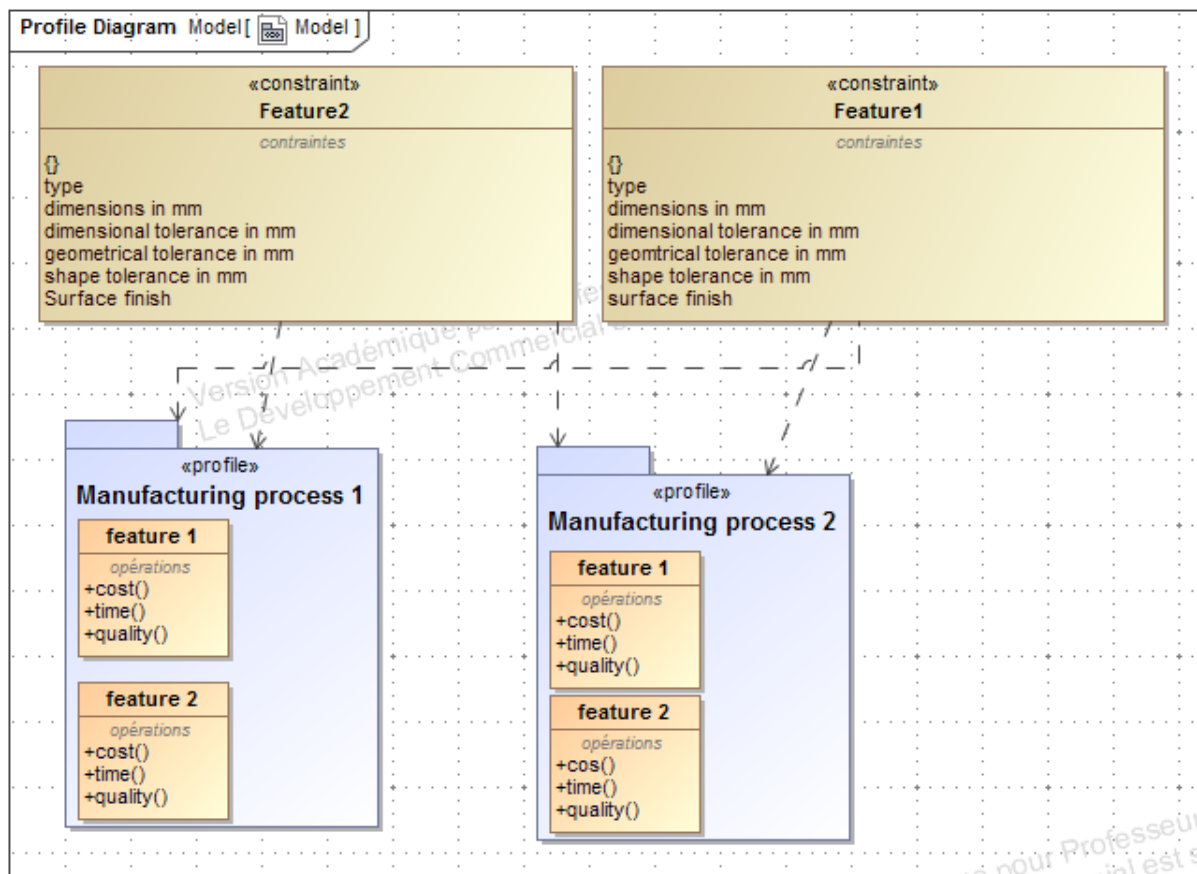


Fig.4.2 : le diagramme de profil SysML développé pour l'étude de la fabricabilité

Ce modèle est développé au moyen d'allocations et de flux de satisfaction entre les spécifications du client, les contraintes de conception de la pièce, les processus de fabrication et les contraintes liées aux ressources d'atelier de fabrication, afin de converger vers la solution qui optimise l'utilisation du temps et des ressources.

La première étape de l'approche proposée consiste à définir et à prendre en compte les besoins des clients et des parties prenantes internes. Cette étape vise à comprendre les spécifications des clients afin de définir les objectifs du système requis. Ces spécifications sont présentées sous formes d'exigences. Celles de niveau supérieur sont décomposées en exigences plus détaillées, puis structurées dans le «diagramme d'exigences SysML». Elles traduisent une capacité ou une contrainte à satisfaire par un système. A partir de ces exigences, les valeurs des clients sont capturées et classées en valeurs singulières.

Afin de mieux comprendre la fonctionnalité du système, une architecture fonctionnelle contenant l'activité principale et les activités des sous-systèmes est intégrée dans le diagramme d'activité SysML.

En se basant sur les valeurs du client et sur l'architecture fonctionnelle, de nombreux cas d'étude du système sont pris en compte dans l'environnement SysML. Pour choisir le meilleur cas d'étude, un raisonnement à partir de cas (CBR) [97] est effectué pour trouver un cas similaire et pour le réutiliser dans la nouvelle situation du problème. Le raisonnement à partir de cas est une approche d'apprentissage incrémental, qui permet d'acquérir une nouvelle expérience chaque fois qu'un problème est résolu, le rendant immédiatement disponible pour les problèmes futurs. Pour implémenter ce paradigme dans le modèle MBMA, une «matrice d'allocation SysML» est utilisée pour intersecter les attributs de valeurs clé des clients avec des contraintes du système retenu.

Dans la deuxième étape, les équipes de sous-systèmes travaillent en collaboration pour définir les exigences fonctionnelles liées à la partie mécanique du système. Elles sont introduites dans le diagramme d'exigences SysML. A partir de ces exigences, les concepteurs proposent un dessin de prototype 3D qui va servir pour l'étude de la fabricabilité.

Dans la troisi me  tape, un flux ascendant issu des concepteurs et des experts en fabrication contenant des informations sur les contraintes extraites du prototype 3D est ensuite impl ment  dans le «diagramme de d finition de blocs SysML » (BDD). Les contraintes sont utilis es comme artefacts principaux pour d crire les st r otypes des entit s de fabrication de cette partie. Par exemple, pour d crire une entit  de fabrication, des contraintes concernant la tol rance et l  tat de surface sont n cessaires pour l analyse de la fabricabilit .

La troisi me  tape de cette approche consiste   d finir le syst me de fabrication disponible dans l atelier de fabrication de l entreprise. Ce syst me est compos  de ressources tels que les machines-outils, les outils de coupe, les porte-outils et les montages possibles. Chaque ressource poss de des contraintes sp cifiques tels que la puissance, la capacit , le co t horaire et la tol rance de chaque machine. Le but de cette  tape est d assister les experts, les ing nieurs de fabrication et les autres parties prenantes lors de la cartographie de l espace de solutions des diff rentes alternatives de processus de fabrication. Par cons quent, un flux ascendant issu des ing nieurs de production contenant les contraintes des ressources est impl ment  dans le mod le MBMA   travers le diagramme de d finition de blocs (BDD) SysML.

Apr s la phase de d finition de contraintes, le lien entre la pi ce et le syst me de fabrication est mis en  uvre dans le diagramme d exigences SysML. En effet, pour assurer une bonne qualit  de la pi ce finie, il faut prendre en compte plusieurs contraintes d allocation entre les ressources telles que les contraintes de compatibilit , d accessibilit  entre la pi ce et le syst me de fabrication.

Pour cartographier l espace de solutions, les parties prenantes utilisent les informations ci-dessus comme entr es pour d finir l ensemble des processus de fabrication en fonction des ressources disponibles dans l atelier de fabrication et les attributs des entit s de fabrication.

Chaque processus de fabrication est affect     la machine-outil correspondante   travers une «matrice d'allocation SysML». Les attributs des processus sont d finis   partir du diagramme de profil SysML d velopp  . L'avantage d'utiliser cette architecture d'allocation est la description compl te de l'ensemble de processus de fabrication et la port e du travail de fabrication requis.

Dans cette phase, le nombre de solutions possibles peut  tre tr s  lev  et les d cisions concernant la solution finale sont difficiles   prendre car elles n cessitent davantage d'informations sur chaque processus de fabrication. Pour limiter l'espace de solutions, nous adoptons deux filtres simultan ment ou progressivement bas s sur la connaissance d'experts et sur une analyse de compromis.

Pour le premier filtre, les directives de la conception pour la fabrication sont utilis es pour supprimer les solutions irr alisables. En fait, ces directives consistent en un ensemble de r gles de conception correspondant   des param tres g om triques pour un processus de fabrication donn e. Une r gle de conception est une recommandation qui garantit que le param tre g om trique correspondant peut  tre fabriqu  correctement. Ces r gles g om triques sont appliqu es au niveau des entit s de fabrication. Si toutes les r gles g om triques pour un processus de fabrication donn  satisfont toutes les contraintes des entit s alors le processus est retenu dans l'espace de solutions. Inversement, si une entit  de fabrication ne peut pas  tre fabriqu e   partir d'un processus de fabrication donn , alors le processus est  limin  de l'espace de solutions. S'il ne reste plus de processus de fabrication apr s l'application du premier filtre, alors la pi ce est consid r e comme non-fabricable et le concepteur doit changer la conception de la pi ce.

Par conséquent, nous obtenons un ensemble réduit de solutions réalisables. Cette étape correspond au troisième principe de l'approche SBCE «Développement d'un ensemble de concept», qui consiste à explorer chaque solution en ajoutant des connaissances et en la communiquant avec les autres équipes afin de recevoir des informations en retour et de comprendre les contraintes. Pour implémenter le premier filtre dans le modèle MBMA, une «matrice de satisfaction SysML» est utilisée pour intersecter des paramètres géométriques dans des lignes avec des processus de fabrication dans des colonnes. Dans cette matrice, chaque ligne présente un filtre de processus pour les paramètres géométriques partagés entre l'ensemble de processus de fabrication. Par conséquent, si un processus satisfait tous les paramètres géométriques, un flux avec un diamant noir apparaît dans la colonne correspondante, sinon un flux avec un diamant blanc apparaît pour indiquer que le processus de fabrication ne respecte pas le paramètre géométrique. L'avantage de cette phase de test réside dans la possibilité d'analyser la capacité d'un ensemble de processus de fabrication pour fabriquer une pièce en temps voulu.

Le deuxième filtre consiste à effectuer une analyse de compromis de la pièce. Un algorithme est développé ainsi pour effectuer une analyse de domaine spécifique dans un environnement SysML, pour réduire l'espace des solutions et pour intégrer une analyse produit / processus. Premièrement, les attributs des compromis contenant la valeur du coût, de la qualité et du temps du processus i pour fabriquer l'entité j sont extraits des diagrammes SysML développés à partir du diagramme de profil. Dans cette phase de développement, les informations sur les attributs de fabricabilité ne sont pas toujours disponibles dans la plate-forme basée sur la connaissance (KB) de l'entreprise. Les experts en fabrication doivent donc intervenir pour classer les valeurs des attributs sur une échelle de 1 à 5.

Cette étape correspond au quatrième principe de SBCE, «Convergence de concept». En fait, dans cette étape, l'élimination a lieu à la lumière de plusieurs activités, notamment l'évaluation des coûts et la convergence progressive vers une solution. La sortie de cet algorithme est un processus de fabrication qui optimise le coût, la qualité et les délais.

Pour modéliser l'espace des solutions et cartographier la structure des diagrammes SysML de cette approche, nous utilisons une formalisation des valeurs du client, des exigences fonctionnelles, des entités de fabrication, des processus de fabrication et des ressources de fabrication avec une structure F , comme suit :

$F = \{R, A, Z, F, S, P, H, E, V, K, G\}$ où:

$R = \{r_1, \dots, r_n\}$ est l'ensemble des valeurs du client que le système doit satisfaire.

$A = \{a_1, \dots, a_b\}$ est l'ensemble des exigences fonctionnelles de la pièce.

$Z = \{z_1, \dots, z_c\}$ est l'ensemble des solutions techniques de la pièce.

$F = \{f_1, \dots, f_m\}$ est l'ensemble d'entités de fabrication de la pièce.

$S = \{s_1, \dots, s_w\}$ est l'ensemble des ressources de fabrication disponibles dans l'atelier.

$P = \{p_1, \dots, p_x\}$ est l'ensemble des processus de fabrication constituant l'espace des solutions.

$H \subseteq (R \times F)$ est l'ensemble des relations entre les valeurs du client et les contraintes de conception de la pièce. Pour chaque valeur $r \in R$, il existe au moins une contrainte de la pièce $f \in F$ tel que $H(r, f)$. Cette relation traduit le premier principe de SBCE, qui est la «recherche de valeur».

$E \subseteq (A \times Z)$ est l'ensemble des relations entre les exigences fonctionnelles de la pièce et les solutions techniques de la pièce. Pour chaque valeur $a \in A$, il existe au moins une solution technique $z \in Z$ tel que $E(a, z)$. Cette relation traduit le taux d'innovation de l'entreprise.

$V \subseteq (F \times S)$ est l'ensemble des relations entre les entités de fabrication et les ressources de fabrication où, pour chaque entité $f \in F$, il existe au moins une ressource $s \in S$ telle que $V(f, s)$. Cette relation traduit l'allocation entre les ressources pour garantir le bon déroulement de l'opération de fabrication.

$K(S \times P)$ est l'ensemble des relations entre les ressources de fabrication et les processus de fabrication où, pour chaque ressource de l'atelier $s \in S$, il existe au moins $p \in P$ tel que $K(s, p)$. Cette relation met en évidence le deuxième principe de SBCE, qui est «cartographier l'espace de conception».

$G \subseteq (F \times P)$ est l'ensemble des relations entre les entités de fabrication et les processus de fabrication où, pour chaque entité $f \in F$, il existe au moins un processus $p \in P$ tel que $G(f, p)$. Cette relation met en évidence les processus de fabrication possibles qui pourraient fabriquer une entité donnée.

Le tableau 6 résume les étapes de l'approche proposée pour intégrer les principes du SBCE dans le modèle MBMA :

Les principes du SBCE	Intégration des principes du SBCE dans le modèle MBMA
Recherche de valeurs	Pour choisir la conception qui correspond aux valeurs du client, un raisonnement à partir de cas est mené afin de répondre aux spécifications du client
Cartographier l'espace de conception	L'espace de conception est remplacé par un ensemble de processus de fabrication. Pour aboutir à ces processus, les contraintes liées à la pièce et aux ressources de fabrication sont considérées pour bien définir l'espace de solutions.
Développement d'un ensemble de concept	Pour tester la capabilité des processus de fabrication, des directives de DFM sont collectées et comparées aux contraintes de la pièce. Les processus qui ne respectent pas les contraintes de la pièce, sont éliminés de l'espace de solutions.
Convergence de concept	Cette étape contient un nombre réduit de solutions. Pour converger vers la meilleure solution, un algorithme est développé pour analyser les compromis de chaque processus.
Conception détaillée	Le meilleur processus qui optimise le coût, le temps et la qualité de la pièce est ensuite communiqué à d'autres ingénieurs métiers.

Tab.6 : Intégration des principes du SBCE dans le modèle MBMA

4.3. Implémentation de l'algorithme

Pour mener une analyse de compromis, nous avons développé un algorithme avec le langage Python qui permet de comparer l'ensemble de solutions réduit et de choisir la meilleure solution qui optimise le coût, le temps et la qualité de la pièce. La première étape de cette analyse, consiste à extraire les attributs des entités de fabrication et des processus de fabrication à partir des diagrammes SysML. Ensuite, l'utilisation de l'outil Gephi permet d'illustrer l'espace de solutions sous forme de graphe comme le montre la figure 4.3 :

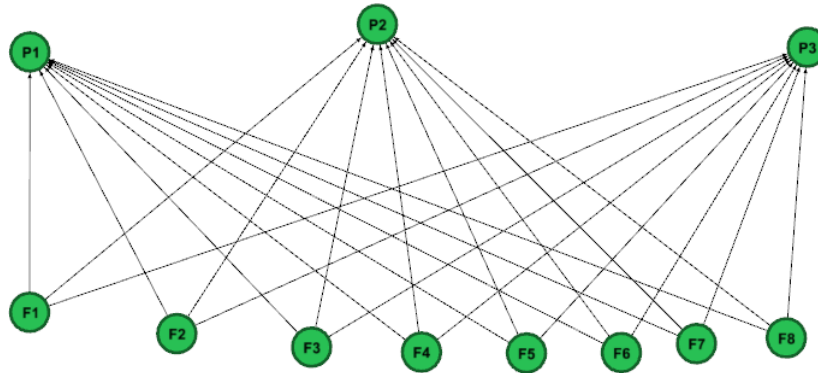


Fig.4.3 : Le graphe initial généré par Gephi pour l'étude de la fabricabilité

Ce graphe contient des nœuds et des arcs. Les nœuds attribués aux processus de fabrication sont libellés par P_i ($i = \{1, \dots, n\}$), tandis que les nœuds attribués aux entités de fabrication sont libellés par F_i ($i = \{1, \dots, m\}$). Les arcs entre les nœuds contiennent des attributs de coût, de temps et de qualité du processus i pour une entité j .

La deuxième étape consiste à exporter le graphe initial vers Python pour converger vers la meilleure solution. L'organigramme de cette analyse est présenté à la figure 4.4.

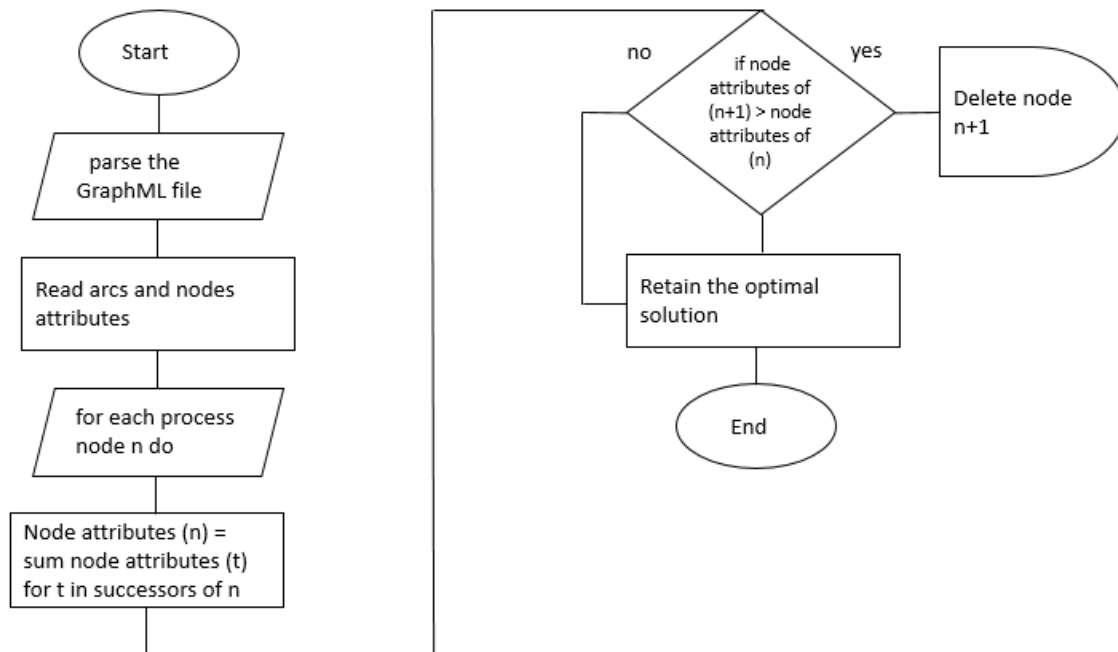


Fig.4.4 : L'organigramme de l'analyse des compromis

L'entrée de cet algorithme est constituée des différents attributs représentés par des arcs entre les processus de fabrication et les entités de fabrication classées de un à cinq. Cinq correspond à la valeur la plus faible donnée par l'expert, tandis que un est la meilleure valeur qui répond au mieux aux exigences du client et des parties prenantes.

L'objectif de cette phase est d'optimiser les valeurs des compromis de l'équation suivante :

$$V(n) = \sum_{i=1}^n T(i) \quad (5)$$

Où n est le nombre de compromis et T(i) est la valeur de chaque compromis. L'agrégation de chaque coût élémentaire, du temps et de la qualité du processus i pour une entité j constitue alors le score total du processus i. L'étape suivante consiste à comparer les nœuds de processus en fonction du score total des compromis.

4.4. Etude de cas d'un boîtier papillon

Pour valider notre approche, une étude de cas d'un corps de boîtier papillon électronique (ETB) est considérée pour l'analyse de la fabricabilité avant la phase d'industrialisation. Le boîtier papillon électronique (ETS) est un dispositif mécatronique situé entre le boîtier de filtre à air et le collecteur d'admission. Il est utilisé pour réguler la quantité d'air dans un moteur à combustion interne. Ce système mécatronique entraîne des améliorations significatives des émissions des véhicules, de l'économie du carburant et de la maniabilité par rapport au boîtier papillon mécanique. En effet, dans les véhicules classiques, l'accélérateur est lié mécaniquement au volet mobile (papillon). Ce système est remplacé dans les véhicules modernes par un dispositif électronique, où la liaison entre la pédale d'accélérateur et le volet mobile est réalisée au moyen d'un servomoteur à courant continu. De cette manière, l'unité de contrôle peut corriger la valeur de la position du volet mobile pour des modes de fonctionnement spécifiques du moteur. Les constituants du boîtier papillon sont présentés dans la figure 4.5 :



Fig.4.5 : le boîtier papillon électronique

Le boîtier papillon électronique est constitué d'un moteur, des engrenages, d'un corps, des capteurs, d'un régulateur de flux d'air et d'un volet mobile (papillon). Pour la conception de ce système mécatronique, les valeurs du client sont présentées dans le diagramme d'exigences SysML de la figure 4.6.

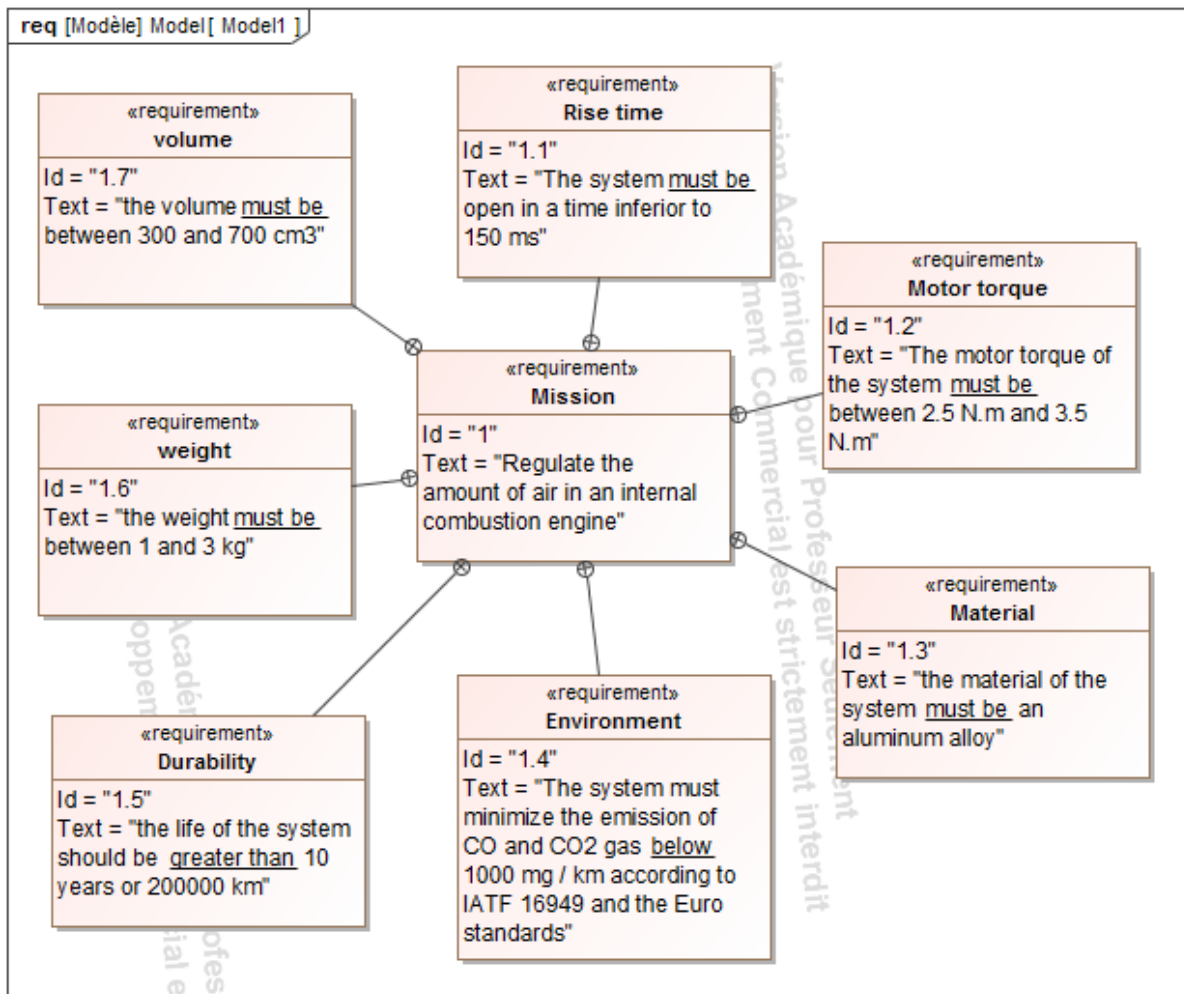


Fig.4.6 : Diagramme d'exigences SysML du boîtier Papillon intégrant les valeurs du client

La première étape de cette approche consiste à définir les valeurs du client. Premièrement, à travers les spécifications du client, les parties prenantes identifient les valeurs principales et les classent en valeurs singulières: couple moteur, matériau de la pièce, impact sur l'environnement, durabilité, poids et temps d'ouverture du volet mobile. Ces valeurs sont présentées en intervalles pour délimiter le nombre de solutions de conception. Afin de mieux comprendre la fonctionnalité de ce système, une architecture fonctionnelle est présentée dans le diagramme d'activité SysML de la figure 4.7 :

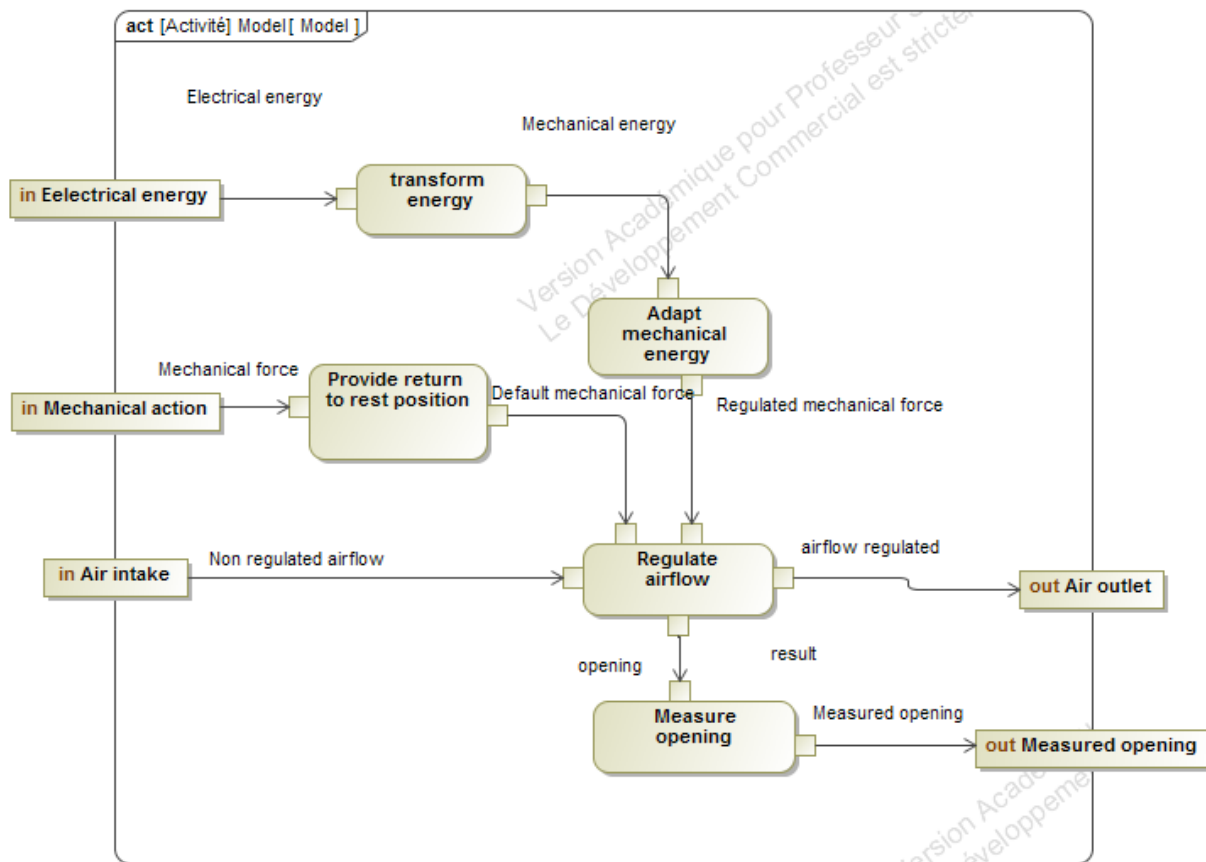


Fig.4.7 : Diagramme d'activité SysML intégrant l'architecture fonctionnelle du boîtier papillon

Dans ce diagramme, l'activité principale du boîtier papillon est décomposée en sous-fonctions. En effet pour réguler la quantité d'air dans un moteur à combustion interne, nous avons besoin d'une unité qui transforme l'énergie électrique en énergie mécanique et une autre unité qui adapte cette énergie mécanique. Ceci construit l'architecture fonctionnelle et permet donc aux parties prenantes de mieux comprendre le fonctionnement de ce système

En se basant sur les valeurs singulières et sur l'architecture fonctionnelle du système, un ensemble de cas d'étude de ce système mécatronique est considéré dans l'environnement SysML comme le montre le tableau 7:

	Cas 1	Cas 2	Cas 3
Emissions de CO et de CO2 (mg/Km)	1200	900	1500

Durabilité (Km)	220000	250000	200000
Matériau	composite	Alliage d'aluminium	fonte
Couple moteur (N.m)	4	3	2
Temps d'ouverture (m.s)	170	140	160
Volume (cm ³)	600	500	400
Poids (Kg)	2	1.5	4

Tab.7 : Cas d'étude des boitiers papillon considérés pour le choix de la meilleure solution

Pour aboutir à la meilleure solution système, un raisonnement à partir de cas est effectué pour converger vers la solution qui répond aux exigences client. En effet, ce raisonnement est effectué par la comparaison des exigences client avec les valeurs des attributs de chaque solution. Dans le tableau 7, nous remarquons que les cas 1 et 3 ne répondent pas aux exigences des clients donc ils sont éliminés. Seulement le cas 2 est retenu pour faire des analyses ultérieures. Pour intégrer ce paradigme dans le modèle MBMA, nous avons utilisé une matrice d'allocation SysML comme le montre la figure 4.8 :

Legend		1.5 kg	3 N.m	140 ms	500 cm ³	900 mg/km	250000 KM	Aluminum alloy
	Allocate							
	Co and CO2 emission					↗		
	durability						↗	
	material							↗
	Motor torque		↗					
	rise time			↗				
	volume				↗			
	weight	↗						

Fig.4.8 : Matrice d'allocation SysML intégrant le cas du boîtier papillon retenu

Pour l'étude de cas du système retenu, nous observons que toutes les contraintes de ce système se situent entre les limites définies par les clients. Par exemple, le couple moteur sélectionné a une valeur de 3N.m, qui correspond aux spécifications du client: «le couple moteur du système doit être compris entre 2,5 N.m et 3,5 N.m».

L'étape suivante consiste à définir les exigences fonctionnelles liées aux corps du boîtier papillon. Ces exigences décrivent les caractéristiques à respecter par cette pièce comme le montre la figure 4.9 :

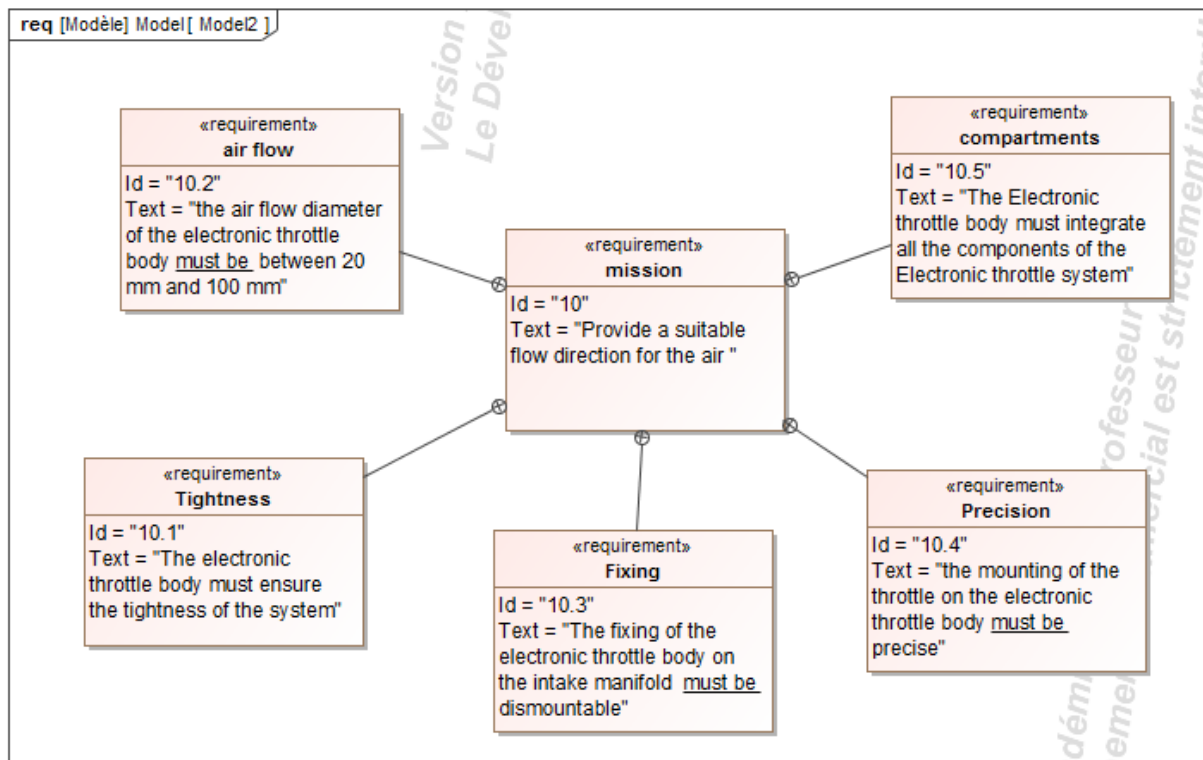


Fig.4.9 : Diagramme d'exigences SysML intégrant les exigences fonctionnelles du corps de boîtier papillon

Dans le diagramme 4.9, le boîtier papillon doit assurer une précision de montage du volet mobile (papillon), une étanchéité du système, un diamètre de passage de l'air entre 20 mm et 100 mm, une fixation démontable sur le collecteur d'admission et des compartiments pour les composants mécanique, électrique et électronique. Ces exigences fonctionnelles serviront de base pour définir les solutions techniques de cette pièce.

En se basant sur ce diagramme d'exigence, les concepteurs traduisent les caractéristiques de corps de boîtier papillon en un prototype de dessin 3D comme le montre la figure 4.10. Ce prototype est établi à travers la collaboration entre les équipes de sous-systèmes et à travers le taux d'innovation requis par l'entreprise.

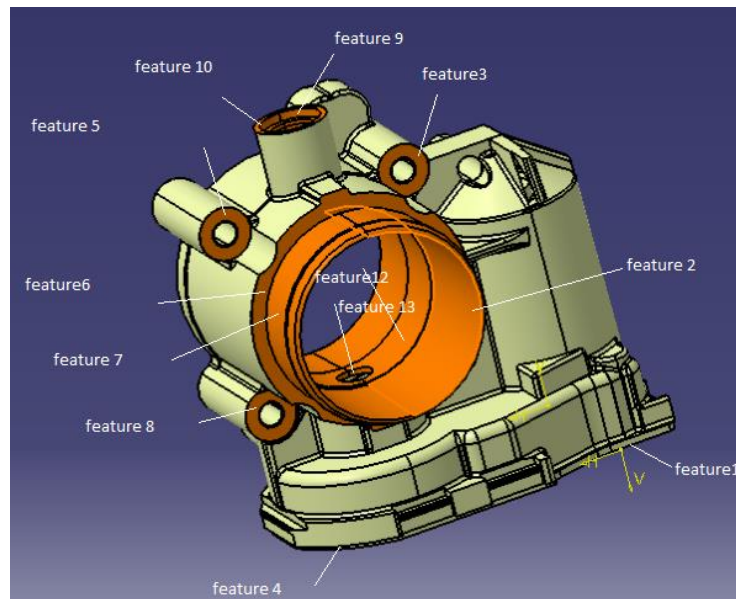


Fig.4.10 : Prototype 3D du boitier papillon

Dans la figure 4.10, les exigences fonctionnelles sont traduites en solution technique du corps de boitier papillon. Par exemple, pour assurer la fixation démontable de cette pièce sur le collecteur d'admission, la solution technique proposée est la fixation par 4 vis. Ceci se traduit par la conception de 4 trous coaxiaux assurant cette exigence. Pour analyser les attributs de ce prototype, les experts extraient les entités de fabrication à partir des surfaces fonctionnelles de la pièce et définissent les attributs de chaque entité : le type de l'entité, les dimensions, les tolérances dimensionnelles, géométrique et de forme et l'état de surface. Durant cette étape de prototypage, l'expert en fabrication définit des critères pour évaluer l'état de surface selon les exigences fonctionnelles du système. Par exemple, pour l'entité 2 (trou), l'état de surface doit être élevé pour assurer l'étanchéité lors de rotation de l'axe de papillon. Les critères d'évaluation de l'état de surface sont définis dans le tableau 8 :

Valeur en μm	Critère
> 6.3	Mauvaise qualité
6.3 – 1.6	Qualité moyenne

1.6 >	Qualité élevée
-------	----------------

Tab.8 : Critère d'évaluation de l'état de surface des entités de fabrication

Ces données sont ensuite implémentées dans le «diagramme de définition de blocs» (BDD), comme illustré à la figure 4.11.

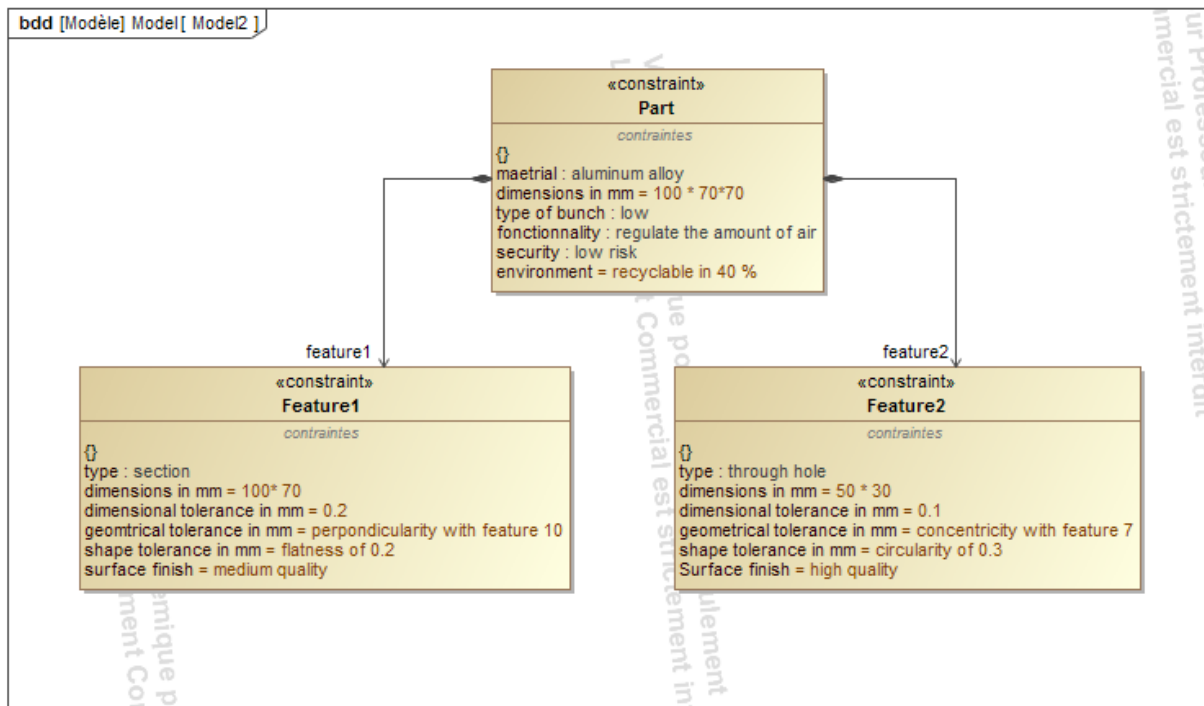


Fig.4.11 : Implémentation des contraintes de conception dans le diagramme de définition de blocs SysML

Dans ce diagramme BDD, la sécurité et l'impact sur l'environnement sont des contraintes supplémentaires pour démontrer que cette pièce automobile est alignée avec la norme IATF 16949, utilisée pour l'échange d'informations entre les ingénieurs de l'industrie automobiles.

L'étape suivante de cette approche consiste à identifier les contraintes des machines-outils disponibles dans l'atelier de fabrication de l'entreprise. En effet, un flux d'échange émis par les ingénieurs de production et contenant les attributs du système de fabrication est implémenté dans le modèle MBMA à travers le diagramme de définition de blocs, comme illustré à la figure 4.12.

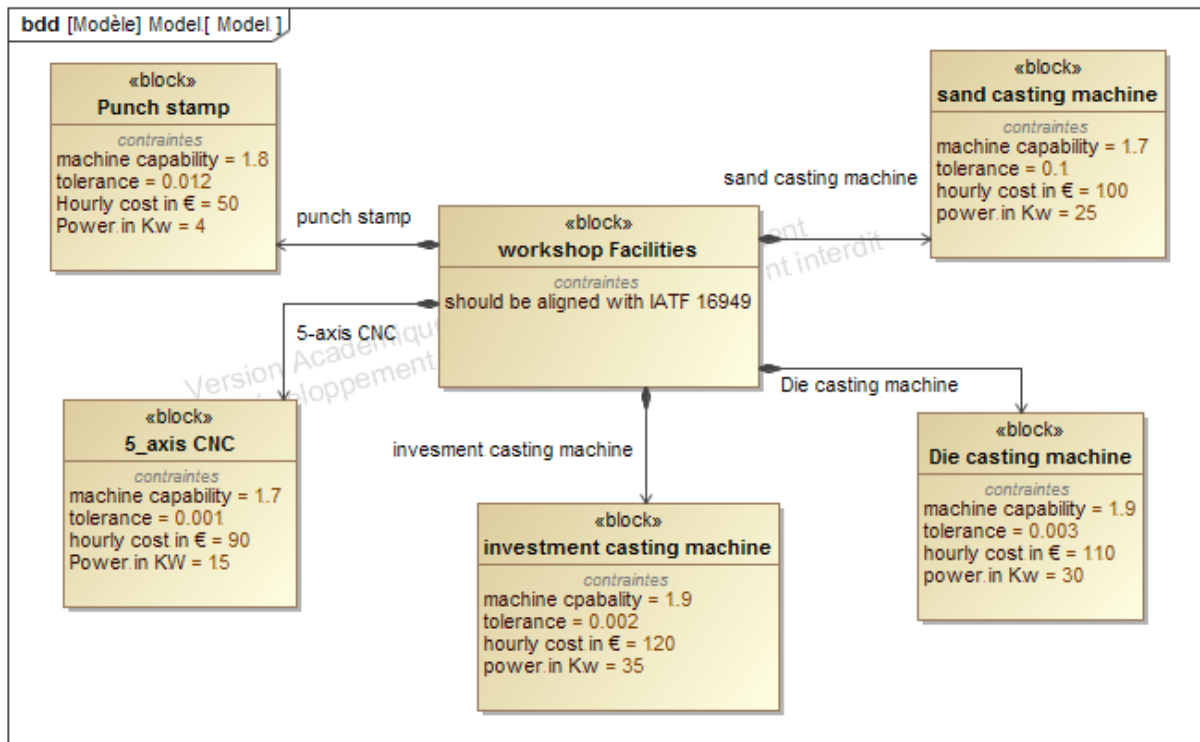


Fig.4.12 : Implémentation des Contraintes des machines-outils dans le diagramme de définition de blocs SysML

Dans ce diagramme, chaque bloc contient des contraintes liées à la machine-outil tels que la capacité machine, la tolérance, la puissance et le coût horaire. Les machines-outils alternatives détectées dans le diagramme BDD sont les suivantes: machine d'estampage, machine à commande numérique 5 axes, machine de moulage de précision, machine de moulage sous pression, machine de moulage en sable.

Pour assurer le bon déroulement des opérations de fabrication entre les machines-outils et la pièce, un diagramme d'exigences SysML est présenté à la figure 4.13 pour mettre en évidence les besoins en allocation de ressources.

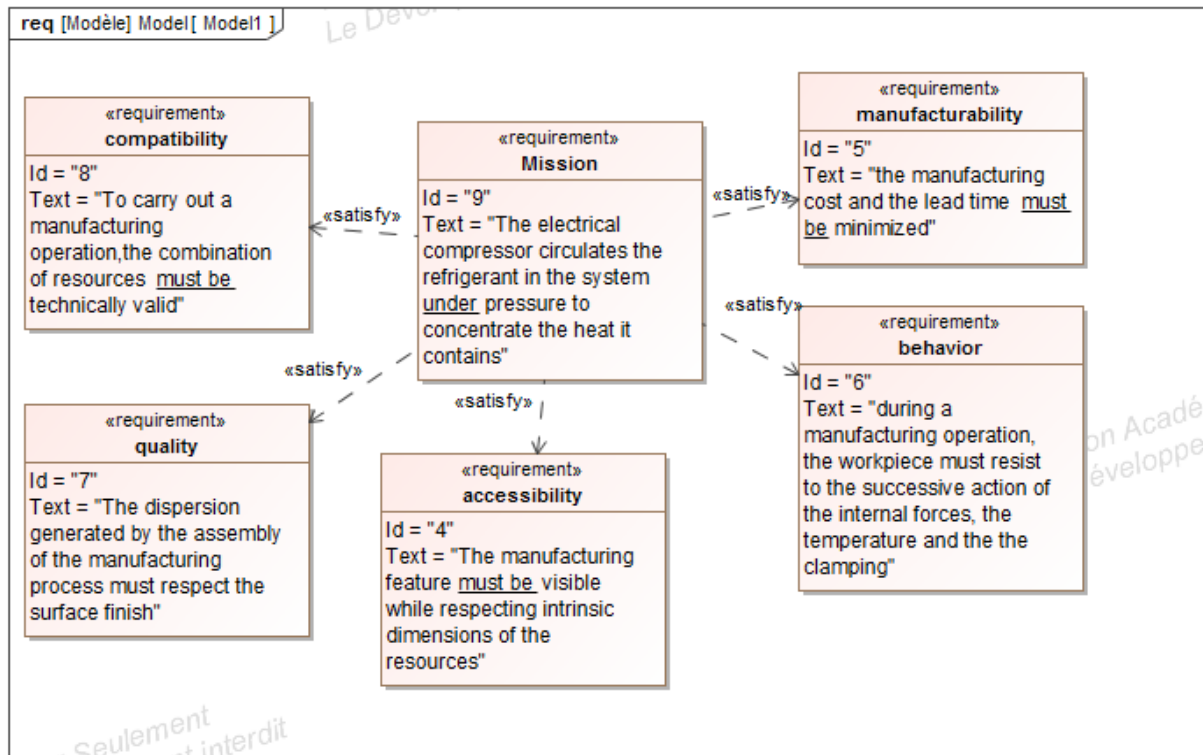


Fig.4.13 : Implémentation des critères d'allocation entre la pièce et la machine-outil dans le diagramme d'exigences SysML

Dans ce diagramme, des critères de qualité, de compatibilité, de comportement, de fabricabilité et d'accessibilité sont détectés afin de mettre en évidence le lien entre la pièce et la machine-outil lors de la fabrication. En effet, ces critères sont essentiels pour garantir la qualité exigée par les clients et par les parties prenantes.

L'étape suivante consiste à cartographier l'espace de solutions. Pour cela, nous utilisons la «Matrice d'allocation SysML» pour allouer à chaque machine-outil le processus de fabrication correspondant, comme illustré à la figure 4.14.

	Estampage	Moulage de précision	Moulage en sable	Usinage	moulage de coulée sous pression
Matériau (alliage d'aluminium)	permis	permis	permis	permis	permis
Les caractéristiques de la pièce (arêtes, dimensions, épaisseurs) en mm	Nombre d'arêtes : ≤ 6 épaisseur : ≥ 0.059 Dimensions : ≥ 0.002	Épaisseur: ≥ 0.025 dimensions: ≤ 50	Épaisseur : ≥ 0.125 Dimensions : ≥ 0.02	Épaisseur: limitée uniquement par la capacité de la machine. Dimensions: ≥ 0.002	épaisseur: ≥ 0.05 diamètre du trou ≥ 0.1 (pour Aluminium)
Tolérance en mm	Tolérance de profil ≥ 0.012	Générale: ± 0.002	Générale: ± 0.1	tournage ± 0.001 , b alésage ± 0.0005 , fraisage ± 0.002 , perçage $\pm 0.008-0.002$,	Générale: ± 0.002
Etat de surface	Qualité moyenne	Bonne qualité	Mauvaise qualité	Bonne qualité	Bonne qualité

Tab.9 : Paramètres géométriques pour l'évaluation de la capacité de l'ensemble des processus de fabrication

Ce tableau présente un ensemble de paramètres géométriques qui sont : le matériau, les caractéristiques de la pièce (nombre d'arêtes, épaisseur et dimensions), la tolérance et état de surface. Ces paramètres géométriques sont considérés comme des filtres pour l'ensemble de processus de fabrication sélectionné. En se basant sur ce tableau ci-dessus, une «matrice de satisfaction SysML» est utilisée dans la figure 4.15 pour réduire l'espace de solution en fonction des processus de fabrication et des contraintes de conception de la pièce.

	Die casting	green sand casting	investment casting	Machining	stamping
Material (aluminum)	✓	✓	✓	✓	✗
Part edges, thickness are	✓	✓	✓	✓	✗
surface finish	✓	✓	✓	✓	✗
Tolerance	✓	✓	✓	✓	✗

Fig.4.15 : Matrice de satisfaction SysML illustrant les filtres utilisés pour l'ensemble des processus de fabrication

Chaque colonne de cette matrice correspond à un processus de fabrication et chaque ligne à un filtre de processus traduisant la capacité de fabrication du processus correspondant pour la liste des paramètres géométriques. Si le processus de fabrication satisfait le paramètre géométrique, un flux avec un diamant noir apparaît dans la matrice, sinon un flux avec un diamant blanc apparaît et le processus correspondant est éliminé. Dans la matrice ci-dessus, le moulage en sable et l'estampage apparaissent avec un diamant blanc dans la deuxième et la cinquième colonne de la matrice donc ces deux processus sont filtrés.

L'étape suivante consiste à adopter une analyse de compromis basée sur les attributs extraits du diagramme BDD développé à partir de diagramme de profil SysML, comme illustré à la figure 4.16:

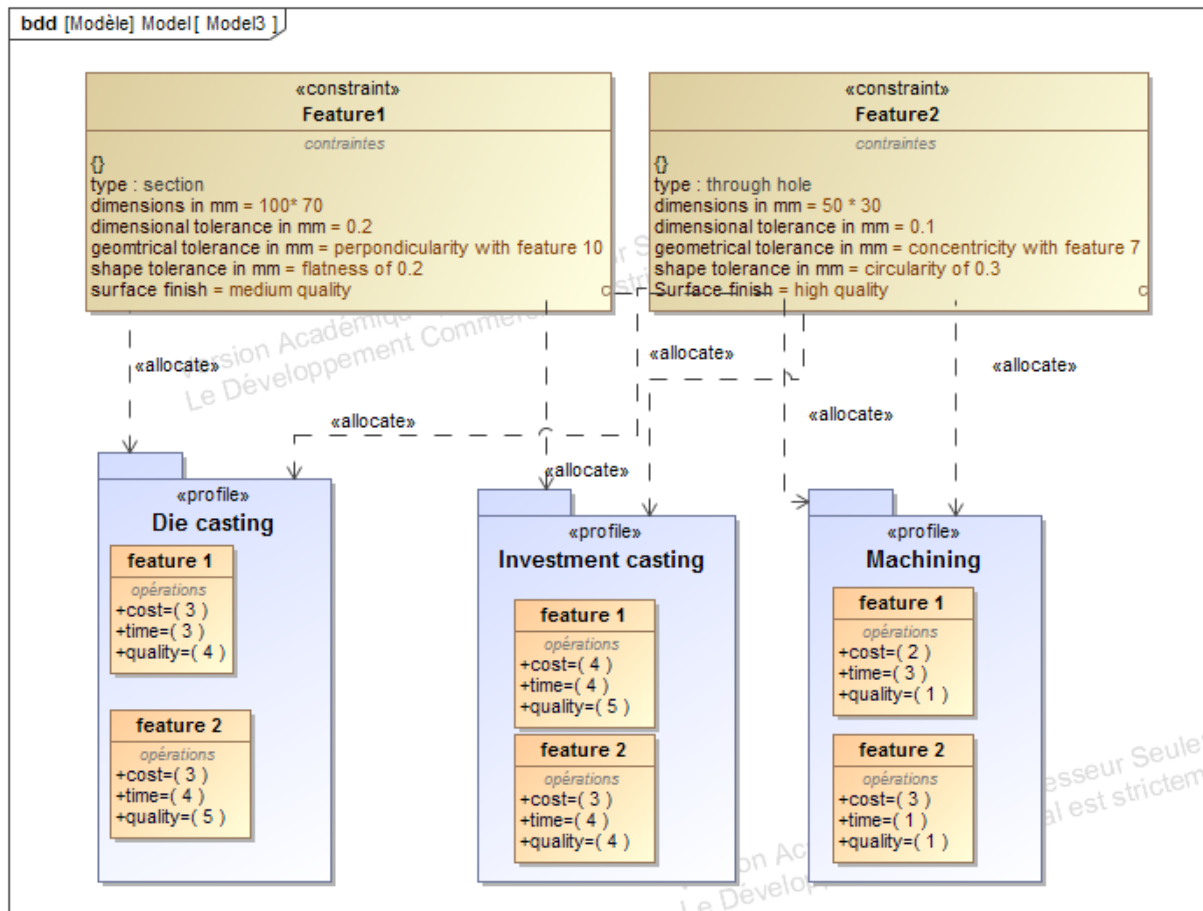


Fig.4.16 : Extraction des attributs du diagramme de définition de blocs SysML

Dans cette étape, une analyse de compromis des valeurs de coût, de temps et de qualité est effectuée pour converger vers la solution de processus optimale. En raison du manque d'informations sur les valeurs des compromis dans la base de connaissance (KB) de l'entreprise, les experts interviennent pour classer les valeurs de coût, de temps et de qualité du processus de fabrication i pour une entité j , sur une échelle de 1 à 5. Dans ce diagramme BDD, quelques entités de fabrication, telles que l'entité 1 (section) et l'entité 2 (trou traversant), sont affectées aux processus restants qui sont : usinage, moulage sous pression et moulage de précision. L'étape suivante consiste à transformer ces informations en un graphe généré par l'outil Gephi. Ceci permet de faciliter l'analyse et la lecture des attributs. Ensuite, un fichier GraphML est exporté vers Python pour converger vers la meilleure solution. Après l'application de la

programmation, le processus de fabrication qui répond le mieux aux besoins du client est l'usinage.

La dernière étape de cette approche consiste à transférer la meilleure solution à d'autres ingénieurs métiers pour faire des analyses de domaine spécifique.

4.5. Discussion

L'étude de cas du boîtier papillon électronique donne un aperçu sur l'application de l'approche SBCE adaptée à la fabrication à travers les diagrammes SysML. Bien que le cadre d'application de cette approche soit dédié au développement de nouveaux produits, il peut être adopté pour l'analyse de la fabricabilité. Ceci nécessite de remplacer l'espace de conception par un ensemble de processus de fabrication qui sont disponibles dans l'atelier de fabrication de l'entreprise.

En Premier lieu, pour répondre aux exigences des clients, un raisonnement à partir de cas est mené dans l'environnement SysML afin de trouver un modèle de cas similaire dans des projets antérieurs. Pour bien mener ce raisonnement, les spécifications des clients sont transformées en valeurs singulières puis implémentées dans le diagramme d'exigences SysML. Ensuite, une architecture fonctionnelle du système est implémentée dans le diagramme d'activité Sysml afin de mieux comprendre la fonctionnalité du système. Ceci permettra de bien définir les valeurs du client et d'éviter les boucles itératives entre les ingénieurs métiers qui peuvent apparaître après la phase d'industrialisation.

En deuxième lieu, pour cartographier l'espace de solutions, les contraintes liées à la pièce et aux ressources de fabrication sont introduites dans les diagrammes SysML afin d'aider les parties prenantes à bien définir l'ensemble des processus de fabrication. L'avantage de cette étape est de permettre aux équipes pluridisciplinaires de travailler ensemble et de partager leurs idées afin d'éviter qu'une erreur de fabrication soit détectée après la phase d'industrialisation.

Ceci permettra donc d' viter des co ts suppl mentaires et d'optimiser l'utilisation du temps et des ressources.

En troisi me lieu, pour filtrer l'espace de solutions et pour converger vers la solution qui r pond au mieux aux exigences des clients et des parties prenantes, deux filtres sont utilis s successivement en se basant sur les param tres g om triques de la conception pour la fabrication et sur une  tude de compromis qui int gre le co t, le temps et la qualit  g n r s par chaque processus.

Pour le premier filtre, les param tres g om triques sont utilis s pour  liminer les processus qui ne r pondent pas aux contraintes de la pi ce. Ceci permettra d' viter les boucles it ratives entre les concepteurs et les ing nieurs de fabrication afin d' valuer la fabricabilit  de la pi ce dans la phase aval de conception. En outre, ces param tres g om triques peuvent assister les concepteurs d butants afin d'am liorer leurs conception et leurs permettent ainsi de capitaliser des connaissances sur la conception pour la fabrication. Cette  tape traduit le 3 me principe de l'approche SBCE « D veloppement d'un ensemble de concept » durant laquelle chaque participant   la conception d veloppe et teste un ensemble de solutions conceptuelles.

Pour le deuxi me filtre, les experts travaillent en collaboration avec les autres parties prenantes pour d finir les valeurs des compromis. Ces valeurs sont ensuite extraites   partir du diagramme de d finition de blocs BDD d velopp    partir du diagramme de profil. En se basant sur ces valeurs, une m thode algorithmique est men e afin d'automatiser le choix de la meilleure solution du processus optimisant le temps, le co t et la qualit  de la pi ce.

La port e de cette approche est limit e   la fabrication de pi ces discr tes. L'analyse des pi ces assembl es n'est pas prise en compte dans ce mod le, ni la combinaison de processus de fabrication visant   am liorer les tol rances g om triques et dimensionnelles de la pi ce. En

effet, le nombre de combinaisons possibles peut  tre trop important vu l'ensemble des processus de fabrication  tudi . Des efforts suppl mentaires sont  galement n cessaires pour obtenir la capacit  de fabrication de chaque combinaison. Ceci sera  tudi  dans des travaux de recherche futurs afin d'am liorer la qualit  de la pi ce.

Conclusion générale

Le développement des produits complexes comme les systèmes mécatroniques est souvent retardé à cause des boucles itératives entre les ingénieurs métiers pour évaluer plusieurs aspects du système comme la fabricabilité, la qualité et la fiabilité. La plupart des travaux de recherche réalisés pour le développement de ces produits reposent sur l'utilisation de l'approche par point. Des problèmes avec une telle approche surviennent lorsque les équipes pluridisciplinaires travaillent ensemble pour évaluer la conception du système. En effet, lorsque la conception passe d'un groupe à l'autre pour une critique de différents points de vue fonctionnels, chaque changement entraîne des modifications et des analyses supplémentaires. En outre, en sélectionnant qu'une seule solution de l'espace de conception, les ingénieurs métiers peuvent converger vers une solution réalisable mais qui n'est pas optimale. Des efforts de recherche ont été réalisés pour intégrer l'approche d'ingénierie concourante basée sur les ensembles SBCE dans le processus de développement des produits complexes. L'analyse de ces travaux de recherche dans le premier chapitre montre l'absence d'application d'une approche systématique des principes du SBCE. Cela augmentera le temps de développement. Pour surmonter ces lacunes, le modèle MBRA développé au 3^{ème} chapitre montre l'application systématique des principes du SBCE en commençant par la définition des valeurs du client jusqu'à la convergence vers un nombre limité de solutions optimales. Ce modèle élimine les solutions qui ne sont pas réalisables progressivement en adoptant des filtres. Le premier filtre est utilisé pour tester les solutions techniques des sous-systèmes en appliquant l'outil AMDEC. Le deuxième filtre consiste à développer un algorithme pour analyser la fiabilité des systèmes conçus. Ceci permet de converger vers un ensemble de solutions réduit et optimisé.

Pour l'analyse de la fabricabilité, des travaux de recherche ont été menés pour réduire les boucles itératives entre le fabricant et le concepteur. Ces travaux se basent sur l'intégration des

systèmes d'évaluation de la fabrication afin d'estimer le coût de fabrication pour une conception donnée. Ces travaux ont été réalisés pour évaluer un seul processus de fabrication qui est dans la plupart des cas l'usinage. Pour remédier aux lacunes de ces travaux, le modèle MBMA développé au 4^{ème} chapitre permet d'évaluer un ensemble de processus de fabrication en tenant compte des contraintes liées à la pièce et aux ressources disponibles dans l'atelier de fabrication. Pour assurer une bonne communication entre les parties prenantes et pour vérifier un ensemble d'alternatives de processus de fabrication, Le modèle MBMA intègre l'MBSE avec les principes du SBCE afin d'assurer la traçabilité des contraintes et de converger vers la meilleure solution dans un délai court. Ce modèle repose sur l'utilisation des directives de DFM pour éliminer les processus de fabrication qui ne peuvent pas fabriquer la pièce donnée. Ensuite, l'application d'un algorithme de compromis permet de converger vers la meilleure solution.

L'intégration de l'MBSE avec les principes du SBCE a des effets positifs sur les performances de développement des produits. Les améliorations sont particulièrement dominantes sur le plan de l'innovation, du coût et de temps de convergence du produit. Cela permettra aux équipes de contourner les processus de développement standard chaque fois que nécessaire et d'éviter l'élimination des solutions faisables du produit aux premières étapes du développement. Le développement d'algorithmes aux différentes étapes de développement (cartographie d'espace, réduction de l'espace de conception, etc.) permettra aux concepteurs de maîtriser la complexité des systèmes mécatroniques. Cependant, un petit nombre d'itérations est encore nécessaire pour la prise de décision basée sur des connaissances des experts.

Pour conclure, l'approche proposée présente l'avantage de réduire les coûts de temps et de développement par rapport aux approches traditionnelles. En appliquant une approche systématique contenant les principes du SBCE, les entreprises seront en mesure de mettre en pratique ce processus d'ingénierie simultanée basée sur les ensembles pour améliorer leurs

systèmes de développement des produits. Pour les travaux futurs, nous pourrions intégrer d'autres disciplines au stade de la conception en amont, tels que la maintenance, l'assemblage et le comportement dynamique, afin d'obtenir plusieurs vues du système. Chaque spécialiste génère un ensemble de solutions faisables dont l'intersection permet de converger vers la meilleure solution.

Référence :

- [1] Lu, Y., Morris, K. C., & Frechette, S. (2016). Current standards landscape for smart manufacturing systems. National Institute of Standards and Technology, NISTIR, 8107, 39.
- [2] Rauch, E., Dallasega, P., & Matt, D. T. (2016). The way from lean product development (LPD) to smart product development (SPD). *Procedia CIRP*, 50, 26-31.
- [3] Yoshimura, M. (2007). System design optimization for product manufacturing. *Concurrent Engineering*, 15(4), 329-343.
- [4] Kartha, C. P. (2004). A comparison of ISO 9000: 2000 quality system standards, QS9000, ISO/TS 16949 and Baldrige criteria. *The TQM magazine*, 16(5), 331-340.
- [5] Giffin, M., de Weck, O., Bounova, G., Keller, R., Eckert, C., & Clarkson, P. J. (2009). Change propagation analysis in complex technical systems. *Journal of Mechanical Design*, 131(8), 081001.
- [6] Jansma, P. A., & Jones, R. M. (2006, January). Advancing the practice of systems engineering at JPL. In *Aerospace Conference, 2006 IEEE* (pp. 19-pp). IEEE.
- [7] International Council on Systems Engineering. (2004). *Systems Engineering Handbook: A" what to" Guide for All SE Practitioners*. INCOSE.
- [8] Bloomberg, J., & Schmelzer, R. (2006). *Service orient or be doomed: how service orientation will change your business*. John Wiley & Sons.
- [9] Jagdev, H. S, Browne, J. (1998). "The extended enterprise-a context for manufacturing." *Production Planning and Control* 9: 216-229.
- [10] Tien, J. M., & Berg, D. (2003). A case for service systems engineering. *Journal of systems science and systems engineering*, 12(1), 13-38.
- [11] Honour, E. C., & Valerdi, R. (2006). Advancing an ontology for systems engineering to allow consistent measurement.
- [12] Electronic Industries Alliance. (1999). *EIA-632: Processes for Engineering a System*.
- [13] Valerdi, R., & Wheaton, M. (2005). ANSI/EIA 632 as a standardized WBS for COSYSMO. In *AIAA 5th ATIO and 16th Lighter-Than-Air Sys Tech. and Balloon Systems Conference* (p. 7373).
- [14] Doran, T. (2006). IEEE 1220: for practical systems engineering. *IEEE Computer*, 39(5), 92-94.
- [15] Büthe, T. (2010). Engineering uncontestedness? The origins and institutional development of the International Electrotechnical Commission (IEC). *Business and Politics*, 12(3), 1-62.
- [16] Rochet, S. (2007). *Formalisation des Processus de l'Ingénierie Systèmes: Proposition d'une méthode d'adaptation des processus génériques à différents contextes d'application* (Doctoral dissertation, Université Paul Sabatier-Toulouse III).
- [17] Ludewig, J. (2003). Models in software engineering—an introduction. *Software and Systems Modeling*, 2(1), 5-14.

- [18] Ramos, A. L., Ferreira, J. V., & Barceló, J. (2012). Model-based systems engineering: An emerging approach for modern systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(1), 101-111.
- [19] Piaszczyk, C. (2011). Model based systems engineering with department of defense architectural framework. *Systems Engineering*, 14(3), 305-326.
- [20] MacCalman, A., Lesinski, G., & Goerger, S. (2016). Integrating External Simulations Within the Model-Based Systems Engineering Approach Using Statistical Metamodels. *Procedia Computer Science*, 95, 436-441.
- [21] Friedenthal, S., Moore, A., & Steiner, R. (2014). A practical guide to SysML: the systems modeling language. Morgan Kaufmann.
- [22] Bahill, A. T., & Botta, R. (2008). Fundamental principles of good system design. *Engineering Management Journal*, 20(4), 9-17.
- [22] Sage, A. P., & Rouse, W. B. (Eds.). (2009). *Handbook of systems engineering and management*. John Wiley & Sons.
- [23] Weiland, K. J., & Holladay, J. (2017, July). Model-Based Systems Engineering Pathfinder: Informing the Next Steps. In *INCOSE International Symposium* (Vol. 27, No. 1, pp. 1594-1608).
- [24] Boehm, B. (2006, May). A view of 20th and 21st century software engineering. In *Proceedings of the 28th international conference on Software engineering* (pp. 12-29). ACM.
- [25] Boehm, B. W. (1988). A spiral model of software development and enhancement. *Computer*, 21(5), 61-72.
- [26] Forsberg, K., & Mooz, H. (1992). The relationship of systems engineering to the project cycle. *Engineering Management Journal*, 4(3), 36-43.
- [27] Forsberg, K., & Mooz, H. (1995, July). 4.4. 4 Application of the 'Vee' to Incremental and Evolutionary Development. In *INCOSE International Symposium* (Vol. 5, No. 1, pp. 848-855).
- [28] Estefan, J. A. (2007). Survey of model-based systems engineering (MBSE) methodologies. *IncoSE MBSE Focus Group*, 25(8), 1-12.
- [29] Cranefield, S., & Purvis, M. K. (1999). UML as an ontology modelling language (pp. 46-53). New Zealand: Department of Information Science, University of Otago.
- [30] Smeda, A., Alti, A., Oussalah, M., & Boukerram, A. (2009). Cosastudio: A software architecture modeling tool. *World Academy of Science, Engineering and Technology*, 49, 263-266.
- [31] Bock, C. (2006). SysML and UML 2 support for activity modeling. *Systems Engineering*, 9(2), 160-186.
- [32] OMG, O. (2007). *Unified Modeling Language: Superstructure 2.1*. 2. Object Management Group (November 2007).
- [33] Hoffmann, H. P. (2008). SysML-based systems engineering using a model-driven development approach. White Paper, Telelogic.

- [34] Roques, P. (2006). UML in practice: the art of modeling software systems demonstrated through worked examples and solutions. John Wiley & Sons.
- [35] Friedenthal, S., Moore, A., & Steiner, R. (2008, June). OMG Systems Modeling Language (OMG SysML™) Tutorial. In INCOSE international symposium (Vol. 18, No. 1, pp. 1731-1862).
- [36] Hoffmann, H. P. (2008). Harmony/SE: A SysML Based Systems Engineering Process. Innovation.
- [37] Cantor, M., & Plug, R. U. P. (2003). Rational Unified Process for Systems Engineering Part 1: Introducing RUP SE Version 2.0. The Rational Edge (August 2003).
- [38] Childers, S. R., & Long, J. E. (2004, August). A concurrent methodology for the system engineering design process. In INCOSE International Symposium (Vol. 4, No. 1, pp. 226-231).
- [39] Hsu, J. C. (2006, July). 4.3. 3 Applying Systems Modeling Language to A Simple Hardware System. In INCOSE International Symposium (Vol. 16, No. 1, pp. 595-605).
- [40] Wang, L., Izygon, M., Okon, S., Wagner, H., & Garner, L. (2016). Effort to Accelerate MBSE Adoption and Usage at JSC. In AIAA SPACE 2016 (p. 5542).
- [41] Mhenni F, Choley JY, Penas O, et al. (2014) A SysML-based methodology for mechatronic systems architectural design. Advanced Engineering Informatics 28(3): 218–231.
- [42] Hammadi, M., Choley, J. Y., Penas, O., Riviere, A., Louati, J., & Haddar, M. (2012, November). A new multi-criteria indicator for mechatronic system performance evaluation in preliminary design level. In Mechatronics (MECATRONICS), 2012 9th France-Japan & 7th Europe-Asia Congress on and Research and Education in Mechatronics (REM), 2012 13th Int'l Workshop on (pp. 409-416). IEEE.
- [43] Hammadi, M., Choley, J. Y., Penas, O., & Riviere, A. (2012, October). Multidisciplinary approach for modelling and optimization of Road Electric Vehicles in conceptual design level. In Electrical Systems for Aircraft, Railway and Ship Propulsion (ESARS), 2012 (pp. 1-6). IEEE.
- [44] Ramos, A. L., Ferreira, J. V., & Barceló, J. (2012). Model-based systems engineering: An emerging approach for modern systems. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 42(1), 101-111.
- [45] Simpkins, P., Kleinholz, A., & Maley, J. (2009). A practical application of MBSE—the automated parking system. In 3rd Asia-Pacific Conf. Syst. Eng., Singapore.
- [46] Soyler, A., & Sala-Diakanda, S. (2010, April). A model-based systems engineering approach to capturing disaster management systems. In Systems Conference, 2010 4th Annual IEEE (pp. 283-287). IEEE.
- [47] Vallhagen J, Madrid J, So" derberg R, et al. (2013). An approach for producibility and DFM methodology in aerospace engine component development. Procedia CIRP 11: 151–156.
- [48] Dal Forno, A. J., Forcellini, F. A., & Bornia, A. C. (2013). Lean product development: benchmarking in Brazilian companies. In 22nd International Congress of Mechanical Engineering, Ribeirão Preto, November 3 (Vol. 7).

- [49] Rossi, M., Kerga, E., Taisch, M., & Terzi, S. (2012, July). Learning methodologies to diffuse lean product development to industries. In IFIP International Conference on Product Lifecycle Management (pp. 287-298). Springer, Berlin, Heidelberg.
- [50] Browning, T. R. (2000). Value-based product development: refocusing lean. In Engineering Management Society, 2000. Proceedings of the 2000 IEEE (pp. 168-172). IEEE.
- [51] Abdalla, H. S. (1999). Concurrent engineering for global manufacturing. *International Journal of Production Economics*, 60, 251-260.
- [52] Wang, J., and Terpenney, J., (2003), "Interactive evolutionary solution synthesis in fuzzy set-based preliminary engineering design," *J. Intell. Manuf.*, 14(2), pp. 153–167.
- [53] Raudberget, D., 2010, "Practical applications of set-based concurrent engineering in industry," *Stroj. Vestn.-J. Mech. Eng.*, 56(11), pp. 685–695.
- [54] Khan, M., Al-Ashaab, A., Doultsinou, A., Shehab, E., Ewers, P., and Sulowski, R., (2011), "Set-Based Concurrent Engineering process within the LeanPPD environment," *Improving Complex Systems Today*, Springer, pp. 433–440.
- [55] Sobek, D. K., Ward, A. C., and Liker, J. K., 1999, "Toyota's principles of set-based concurrent engineering," *Sloan Manage. Rev.*, 40(2), pp. 67–84.
- [56] Clarkson, P. J., Simons, C., and Eckert, C., 2004, "Predicting change propagation in complex design," *J. Mech. Des.*, 126(5), pp. 788–797.
- [57] Khan, M., Al-Ashaab, A., Doultsinou, A., Shehab, E., Ewers, P., & Sulowski, R. (2011). Set-based concurrent engineering process within the LeanPPD environment. In *Improving Complex Systems Today* (pp. 433-440). Springer, London.
- [58] Madhavan K, Shahan D, Seepersad CC, et al. (2008). An industrial trial of a set-based approach to collaborative design. In: *Proceedings of the ASME 2008 international design engineering technical conferences and computers*
- [59] Borchani, M. F., Ammar, R., Hammadi, M., Choley, J. Y., Yahia, N. B., Barkallah, M., & Louati J. (2018, September). Mechatronic System Design using Model-Based Systems engineering and Set-Based Concurrent Engineering Principles. In *2018 12th France-Japan and 10th Europe-Asia Congress on Mechatronics* (pp. 32-38). IEEE.
- [60] Pedersen, R., Kvist, M., & Mortensen, N. H. (2005). Method for alignment of product and production concepts. In *DS 35: Proceedings ICED 05, the 15th International Conference on Engineering Design*, Melbourne, Australia, 15.-18.
- [61] Muhd Ikmal I, Bin Mohd M, Al-Ashaab A, et al. (2017) the set-based concurrent engineering application: a process of identifying the potential benefits in the surface jet pump case study. *Procedia CIRP* 60; 350–355.
- [62] Kerga E, Rossi M, Taisch M, et al. (2014). A serious game for introducing set-based concurrent engineering in industrial practices. *Concurrent Engineering* 22(4): 333–346

- [63] Al-Ashaab A, Golob M, Attia UM, et al. (2013). The transformation of product development process into lean environment using set-based concurrent engineering: a case study from an aerospace industry. *Concurrent Engineering* 21(4): 268–285.
- [64] Nahm YE and Ishikawa H. (2005). Representing and aggregating engineering quantities with preference structure for set-based concurrent engineering. *Concurrent Engineering* 13(2): 123–133.
- [65] Bagola, B. (2004). L'américanisation de la langue française sur Internet? Quelques aspects de la terminologie officielle et de l'usage des internautes. *GLOBE: Revue internationale d'études québécoises*, 7(2), 101-124.
- [66] Gupta, S. K., Das, D., Regli, W. C., & Nau, D. S. (1995). Current trends and future challenges in automated manufacturability analysis.
- [67] Gupta, S. K., & Nau, D. S. (1995). Systematic approach to analyzing the manufacturability of machined parts. *Computer-Aided Design*, 27(5), 323-342.
- [68] Bralla, J. G. (1999). *Design for manufacturability handbook* (Vol. 11). New York: McGraw-Hill.
- [69] Medani, O., and Ratchev, S. M. (2006). "A STEP AP224 agent-based early manufacturability assessment environment using XML," *Int. J. Adv. Manuf. Technol.*, 27(9- 10), pp. 854–864.
- [70] Han, J., Kang, M., and Choi, H. (2001). "STEP-based feature recognition for manufacturing cost optimization," *Comput.-Aided Des.*, 33(9), pp. 671–686.
- [71] Pham, D. T., and Ji, C. (1999), "A concurrent design system for machined parts," *Proc. Inst. Mech. Eng. Part B J. Eng. Manuf.*, 213(8), pp. 841–846.
- [72] Chen, Y.-M., Wen, C.-C., and Ted Ho, C. (2003), "Extraction of geometric characteristics for manufacturability assessment," *Robot. Comput.-Integr. Manuf.*, 19(4), pp. 371–385
- [73] Lockett, H. L., and Guenov, M. D. (2005). "Graph-based feature recognition for injection molding based on a mid-surface approach," *Comput.-Aided Des.*, 37(2), pp. 251– 262.
- [74] Korosec, M., Balic, J., and Kopac, J. (2005). "Neural network based manufacturability evaluation of free form machining," *Int. J. Mach. Tools Manuf.*, 45(1), pp. 13–20.
- [75] Mahajan, P. V. (1993). "Automated manufacturability evaluation and strip layout of feature-based stamped part designs."
- [76] Maropoulos, P. G., Bramall, D. G., and McKay, K. R. (2003). "Assessing the manufacturability of early product designs using aggregate process models," *Proc. Inst. Mech. Eng. Part B J. Eng. Manuf.*, 217(9), pp. 1203–1214.
- [77] Gupta, S. K. (1997). "Using manufacturing planning to generate manufacturability feedback," *J. Mech. Des.*, 119(1), pp. 73–80.
- [78] Balasubramaniam, M., Sarma, S. E., & Marciniak, K. (2003). Collision-free finishing toolpaths from visibility data. *Computer-Aided Design*, 35(4), 359-374.
- [79] Rangarajan, A., Radhakrishnan, P., Moitra, A., Crapo, A., & Robinson, D. (2013, August). *Manufacturability analysis and design feedback system developed using semantic framework.*

International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (pp. V004T05A001-V004T05A001). American Society of Mechanical Engineers.

[80] Chen†*, C. H., Occena, L. G., & Fok, S. C. (2001). CONDENSE: a concurrent design evaluation system for product design. *International Journal of Production Research*, 39(3), 413-433.

[81] Frank, M. C., Wysk, R. A., & Joshi, S. B. (2006). Determining setup orientations from the visibility of slice geometry for rapid computer numerically controlled machining. *Journal of manufacturing science and engineering*, 128(1), 228-238.

[82] de Sam Lazaro, A., Engquist, D. T., & Edwards, D. B. (1993). An intelligent design for manufacturability system for sheet-metal parts. *Concurrent Engineering*, 1(2), 117-123.

[83] Venkatachalam, A., Mellichamp, J. M., & Miller, D. M. (1993). A knowledge-based approach to design for manufacturability. *Journal of Intelligent Manufacturing*, 4(5), 355-366.

[84] Prasad, S., Zacharia, T., & Babu, J. (2008). Design for manufacturing (DFM) approach for Productivity Improvement in Medical Equipment Manufacturing. *International Journal of Emerging Technology and Advanced Engineering*, 4(4), 79-85.

[85] Molcho, G., Zipori, Y., Schneor, R., Rosen, O., Goldstein, D., & Shpitalni, M. (2008). Computer aided manufacturability analysis: Closing the knowledge gap between the designer and the manufacturer. *CIRP Annals-Manufacturing Technology*, 57(1), 153-158.

[86] Bidkar, R. A., & McAdams, D. A. (2010). Methods for automated manufacturability analysis of injection-molded and die-cast parts. *Research in Engineering Design*, 21(1), 1.

[87] Jacob, D. V., Ramana, K. V., & Rao, P. V. M. (2004). Automated manufacturability assessment of rotational parts by grinding. *International journal of production research*, 42(3), 505-519.

[88] Amalnik, M. S., El-Hofy, H. A., & McGeough, J. A. (1998). An intelligent knowledge-based system for wire-electro-erosion dissolution in a concurrent engineering environment. *Journal of Materials Processing Technology*, 79(1-3), 155-162.

[89] Standard, M. (1980). Procedures for performing a failure mode, effects and criticality analysis. Department of Defense, Washington, DC, Standard No. MIL-STD-1629A.

[90] Ericson, C. A. (2015). Hazard analysis techniques for system safety. John Wiley & Sons.

[91] Bastian, M., Heymann, S., & Jacomy, M. (2009). Gephi: an open source software for exploring and manipulating networks. *Icwsn*, 361-362.

[92] Naidu, M., Nehl, T. W., Gopalakrishnan, S., & Wurth, L. (2003, October). A semi-integrated sensorless PM brushless drive for a 42 V automotive HVAC compressor. In *Industry Applications Conference, 2003. 38th IAS Annual Meeting. Conference Record of the* (Vol. 3, pp. 1435-1442). IEEE.

[93] Prasad, B., & Rogers, J. (2005, January). A knowledge-based system engineering process for obtaining engineering design solutions. In *ASME 2005 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* (pp. 477-488). American Society of Mechanical Engineers.

- [94] Prasad B and Rogers J. (May, 2009). An iterative decision-based technique for finding topologically feasible solution sets quickly. In: 50th AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics, and materials conference 17th AIAA/ASME/ AHS adaptive structures conference 11th AIAA, Palm Springs, CA, 4–7, p. 2234.
- [95] Prasad B. (1996). Concurrent Engineering Fundamentals Vol I: and Integrated Product Process Organization. 1st ed. Upper Saddle River, NJ: Prentice Hall.
- [96] Miller, L. C. (1993). Concurrent engineering design: Integrating the best practices for process improvement. Society of manufacturing engineers.
- [97] Bernal, M., Haymaker, J. R., & Eastman, C. (2015). On the role of computational support for designers in action. Design Studies, 41, 163-182.
- [98] Boothroyd, G., Dewhurst, P., & Knight, W. A. (2001). Product Design for Manufacture and Assembly, revised and expanded. CRC press.

Liste des publications:

Jet 2016 : Etude comparative des contraintes de modélisation SysML et celles données par l'ISO TS 16949 : Cas d'usinage des pièces prismatiques

CMSM 2017: A decision support system for the choice of machining strategies of mechanical parts by combining SysML diagrams with neuronal networks

ISSE 2017: The Usage of system engineering approach for integrating machining constraints in the upstream design stage

ICAV 2018: A novel approach for reducing the design space of a mechatronic system using Set Based Concurrent Engineering principles

Mechatronics 2018: Mechatronic System Design using Model-Based System Engineering and Set-Based Concurrent Engineering Principles

CERA 2018: Integrating Model-Based System Engineering with Set-Based Concurrent Engineering principles for reliability and manufacturability analysis of mechatronic products

Annexe:

Pour le développement des codes de programmation dans le 3^{ème} et 4^{ème} chapitre, le cadre adapté est Pycharm. En effet, PyCharm est un environnement de développement utilisé pour programmer en Python. Il offre la gestion des tests unitaires, l'analyse de code et l'intégration de logiciel de gestion de versions

1^{er} code : analyse de la fiabilité

```
# create the graph
import matplotlib.pyplot as plt
import networkx as nx

graph=nx.read_graphml("C:/Users/PlacisAdmin/Desktop/gephi/faulttree1.graphml", node_type=str)
frozen_graph = nx.freeze(graph)
G = nx.Graph(frozen_graph)
nx.draw(G, with_labels=True)
plt.show()
# calculate probability of events
for n in G.nodes():
    if G.node[n]['label'] in
['11','12','13','14','21','22','23','24','31','32','33','34','41','42','43',
'44','55','66','77','88']:
        if G.node[n]['gate or']== 1.0:
            times = 1
            somme = 0
            for t in G.adj._atlas[n].keys():
                if G.nodes[t]['label'] not in ('1','2','3','4'):
                    times = times * G.nodes[t]['probabilite']
                    somme += G.node[t]['probabilite']
            G.node[n]['probabilite']= somme - times

        elif G.node[n]['gate and']== 1.0:
            for t in G.adj._atlas[n].keys():
                G.node[n]['probabilite']= G.node[n]['probabilite'] *
G.nodes[t]['probabilite']
            else:
                G.node[n]

# calculate probability of systems
for n in G.nodes():
    if G.node[n]['label'] in ['1','2','3','4']:
        if G.node[n]['gate or']==1.0:
            times = 1
            somme = 0
            for t in G.adj._atlas[n].keys():
                if G.nodes[t]['label'] not in ('E','1','2','3','4'):
                    times = times * G.node[t]['probabilite']
                    somme += G.node[t]['probabilite']

            G.node[ n ][ 'probabilite' ] = somme - times
```

```

        elif G.node[n]['gate and']== 1.0:
            prod = 1
            for t in G.adj._atlas[n].keys():
                if G.nodes[t]['label'] not in ('1', '2', '3', '4'):
                    prod = prod * G.nodes[t]['probabilite']
            G.node[n]['probabilite']=prod
        else:
            G.node[ n ]
for n in G.nodes():
    if G.node[n]['label']in ['1','2','3','4']:
        print(G.node[n]['probabilite'])
# eliminate nodes
x = G.nodes().keys()
for d in x:
    if G.node.get(d) and G.node[d]['label']in ['1','2','3','4'] and
G.node[d]['probabilite'] >a:
        for t in frozen_graph.succ[d].keys():
            for z in frozen_graph.succ[t].keys():
                if G.node.get(z) :
                    G.remove_node(z)
            if G.node.get(t):
                G.remove_node(t)
        #Delete the actual one
        G.remove_node(d)
# show the graph
nx.draw(G, with_labels=True)
plt.show()
print ('Done')

```

2ème code: Diagramme Radar

```

# Libraries
import matplotlib.pyplot as plt
import pandas as pd
from math import pi

# Set data
df = pd.DataFrame ({
    '11': [ 'performance', 'cost', 'durability', 'reliability' ],
    'performance': [],
    'reliability': [],
    'durability': [],
    'Cost': []
})

# ----- PART 1: Create background

# number of variable
categories = list (df)[ 1: ]
N = len (categories)

# What will be the angle of each axis in the plot? (we divide the plot /
number of variable)
angles = [ n / float (N) * 2 * pi for n in range (N) ]
angles += angles[ :1 ]

```

```
# Initialise the spider plot
ax = plt.subplot (111, polar=True)

# If you want the first axis to be on top:
ax.set_theta_offset (pi / 2)
ax.set_theta_direction (-1)

# Draw one axe per variable + add labels labels yet
plt.xticks (angles[ :-1 ], categories)

# Draw ylabels
ax.set_rlabel_position (0)
plt.yticks ([ 1, 3, 5 ], [ "1", "3", "5" ], color="grey", size=7)
plt.ylim (0, 5)

# ----- PART 2: Add plots

# Plot each individual = each line of the data
# I don't do a loop, because plotting more than 3 groups makes the chart
unreadable

# Ind1
values = df.loc[ 0 ].drop ('11').values.flatten ().tolist ()
values += values[ :1 ]
ax.plot (angles, values, linewidth=1, linestyle='solid', label="solution
1")
ax.fill (angles, values, 'b', alpha=0.1)

# Ind2
values = df.loc[ 1 ].drop ('11').values.flatten ().tolist ()
values += values[ :1 ]
ax.plot (angles, values, linewidth=1, linestyle='solid', label="solution
2")
ax.fill (angles, values, 'r', alpha=0.1)

# Ind 3
values = df.loc[ 2 ].drop ('11').values.flatten ().tolist ()
values += values[ :1 ]
ax.plot (angles, values, linewidth=1, linestyle='solid', label="solution
3")
ax.fill (angles, values, 'h', alpha=0.1)

# Add legend
plt.legend (loc='upper right', bbox_to_anchor=(0.1, 0.1))
plt.show ()
```

3ème code : Etude des compromis pour l'analyse de la fabricabilité

```
import matplotlib.pyplot as plt
import networkx as nx
from math import pi

# create the graph
graph = nx.read_graphml ("C:/Users/PlacisAdmin/Desktop/gephi/MBMA.graphml",
node_type=str)
frozen_graph = nx.freeze (graph)
G = nx.Graph (frozen_graph)
nx.draw (G, with_labels=True)
```

```
plt.show ()

#change node attributes
for n in G.nodes():
    if G.node[n]['label']in ['P1','P2','P3','P4']:
        G.node[ n ][ 'tradeoffs ' ] = float (sum (G.edges[n,t]['cost'] for
t in G.adj._atlas[ n ].keys ())) + float (sum (G.edges[n,t]['time'] for t in
G.adj._atlas[ n ].keys ())) + \
                                float (sum (G.edges[n,t]['cost'] for
t in G.adj._atlas[ n ].keys ()))
# SHOW ATTRIBUTES
for n in G.nodes():
    if G.node[n]['label']in ['P1','P2','P3','P4']:
        print(G.node[ n ][ 'tradeoffs ' ])
# eliminate nodes
x = G.nodes().keys()
for d in x:
    for next in range(x.index(d)+1, len(x)):
        if G.node.get(d) and G.node[d]['label']in ['P1','P2','P3','P4'] and
G.node.get(x[next]) \
            and G.node[x[next]]['label']in ['P1','P2','P3','P4']:
            if G.node[d]['tradeoffs'] < G.node[x[next]]['tradeoffs']:
                for t in frozen_graph.succ[x[next]].keys():
                    if G.node.get(t) :
                        G.remove_node(t)
                #Delete the actual one
                G.remove_node(x[next])
            else :
                for t in frozen_graph.succ[d].keys():
                    if G.node.get(t) :
                        G.remove_node(t)
                #Delete the actual one
                G.remove_node(d)
                break

# show the graph
nx.draw (G, with_labels=True)
plt.show ()
print 'Done'
```