Commande sûre de fonctionnement et diagnostic des systèmes à événements discrets : vers une intégration de l'humain au coeur de l'industrie du futur

Philippot Alexandre

UNIVERSITE DE REIMS CHAMPAGNE-ARDENNE

ÉCOLE DOCTORALE SCIENCES DU NUMÉRIQUE ET DE L'INGÉNIEUR

HABILITATION A DIRIGER DES RECHERCHES

Présentée par

Alexandre PHILIPPOT

Maître de Conférences

Docteur de l'Université de Reims Champagne-Ardenne

COMMANDE SÛRE DE FONCTIONNEMENT ET DIAGNOSTIC DES SYSTEMES A EVENEMENTS DISCRETS : VERS UNE INTEGRATION DE L'HUMAIN AU COEUR DE L'INDUSTRIE DU FUTUR

Soutenue publiquement le 03 décembre 2018, devant le jury composé de :

Rapporteurs	
M. Pascal Berruet	Professeur, Lab-STICC – Univ. Bretagne Sud
M. Dimitri Lefebvre	Professeur, GREAH – Univ. Le Havre
M. Frédéric Vanderhaegen	Professeur, LAMIH – Univ. Valenciennes
Examinateurs	
M. Jean-Jacques Lesage	Professeur, LURPA – ENS-Paris-Saclay
Mme Véronique Carré-Ménétrier	Professeur, CReSTIC – Univ. Reims Champagne-Ardenne
M. Bernard Riera	Professeur, CReSTIC – Univ. Reims Champagne-Ardenne (Garant URCA)

REMERCIEMENTS

Ce mémoire d'habilitation synthétise mon activité d'enseignement et de recherche depuis ma titularisation au poste de maître de conférences en 2007.

Pour leur travail de rapporteur et d'examinateur, je tiens à remercier les membres du jury : Pascal Berruet, Dimitri Lefebvre, Frédéric Vanderhaegen pour leurs remarques, conseils et encouragements autour de ce document, et Jean-Jacques Lesage qui depuis longtemps m'a soutenu dans cette démarche.

Je tiens à remercier plus particulièrement Véronique Carré-Ménétrier et Bernard Riera. Merci Véronique pour l'accompagnement que tu fais au quotidien, pour ton sens du détail et ta diplomatie. Merci à mon garant Bernard avec qui je travaille également depuis le début, pour ta confiance en moi et pour cette influence que tu as eu sur ce projet de recherche tourné vers l'opérateur humain et pour ta vision de l'industrie.

Cependant, la rédaction de ce manuscrit n'aurait pas été possible non plus sans mes autres collègues du département ATS du CReSTIC et du département EEA. Tout d'abord mon collègue de bureau, mon binôme et presque mon jumeau numérique… François. Le travail que tu fournis en pédagogie est énorme et devrait être plus reconnu. Promis, tu pourras rester encore en face de moi encore pas mal de temps. Merci à Nadhir, Noureddine, David, Alban, Abdelouahed, Ramla, Pascale avec qui j'encadre ou j'ai pu encadrer nos doctorants. Ce ne sont que quelques poussières laissées mais qui grandissent bien vite. Je n'oublie pas Janan qui m'a incité à effectuer un DEA. Je joins à ces remerciements l'équipe pédagogique qui œuvre énormément à la mise en place des formations : Maxime, Damien, David, Olivier, Danielle, Kevin… Et Laurent pour sa bonne humeur, merci voisin de bureau. Merci également à notre Président Guillaume Gellé qui m'a fait confiance dans le cadre de la mission « Industrie du Futur ».

Je continuerai ces remerciements par les indispensables : nos étudiants/doctorants. Sans eux, pas de formations, pas d'encadrements. Etudiants de Master, ce fût un plaisir d'échanger avec vous et de vous avoir vu progresser dans votre carrière professionnelle (Alexandre, Thibault, Jean-Marie, Arnaud, Jessy …). A mes doctorants à qui je dois bien aussi quelques

mots. Mathieu, qui fût le premier, à quand un autre match de rugby ensemble au stade de France ? Raphaël, Vincent et Mohamed, que j'ai eu également en Master, quand aurais-je enfin le droit à un billet SNCF et un Pass Navigo gratuit ? Yassine, pour le gros travail réalisé dans la dernière année de doctorat : ça a payé. Romain, que j'aurai bien imaginé collègue EC, merci à toi et aux échanges fructueux que l'on a pu avoir. Tu as choisi le cidre au champagne, à chacun ses bulles mais j'espère réellement que l'on continuera à travailler ensemble malgré la distance. Enfin mes petits derniers que je vais encore harceler quelques temps, Imane et Mehdi, arrêtez de lire ces remerciements et allez rédiger votre mémoire ;).

Je ne peux oublier également l'un des sièges que j'ai occupé quelques temps. Sans eux, il serait difficile pour un laboratoire ou un département d'enseignement de fonctionner. Aux anciens et actuels ingénieurs, administratifs et personnels techniques, un big THANKS pour votre aide : Sarah, Sylvie, Nico, Dom, Ma(u)rielle, Daniella, Olivier, Stéphane et Mon Ida. Merci Ségolène pour le suivi des démarches à cette nouvelle ED SNI.

Certains d'entre vous sont passés du cercle professionnel au cercle amical, mais il reste encore un petit cercle familial que je dois remercier. Pour m'avoir supporté et supporté, m'avoir attendu, aidé, relu, écouté, s'être occupé de nos 2 A(s) : Gros bisous ma puce et merci mes enfants. Merci à Mo, Lu, Véro et Michel qui même s'ils ne comprennent pas tout, sont toujours à mes côtés.

Une dernière pensée nostalgique aux LAMPINs. Nos soirées et repas me manquent.

" La Science ne sert qu'à vérifier les découvertes de l'instinct " (Jean Cocteau)

TABLE DES MATIERES

Liste des Figures et Tableaux

Chapitre 1

Chapitre 2

Chapitre 3

Tableaux

INTRODUCTION GENERALE

Ce mémoire a été rédigé en vue d'obtenir l'Habilitation à Diriger des Recherches. Il est pour moi l'occasion de faire une synthèse de mes activités pédagogiques, administratives et de recherche depuis mon recrutement en tant que maître de conférences au sein de l'Université de Reims Champagne Ardenne (URCA). Depuis septembre 2007, j'enseigne principalement au sein des formations portées par le Département EEA de l'UFR Sciences Exactes et Naturelles, et effectue ma recherche au Centre de Recherche en Sciences et Technologies de l'Information et de la Communication (CReSTIC, EA 3804).

Notre statut d'Enseignant/Chercheur représente une exception dans notre société qui implique de trouver un équilibre dans nos trois missions de service public de l'enseignement supérieur. Notre rôle d'Enseignant est de produire et **diffuser le savoir** à nos étudiants dans notre discipline. En tant que Chercheur, nous devons **contribuer à la réflexion** de notre société, à y proposer des solutions à différents défis technologiques. Par ailleurs, il est également de notre devoir d'**accompagner**, d'encadrer nos formations ou nos collègues au travers de Responsabilités collectives. Ces trois missions doivent s'accompagner du trio « Savoir, Savoir-faire, Savoir-être » représentant respectivement la connaissance, la pratique et les attitudes à avoir et à diffuser aussi bien localement que nationalement ou internationalement.

Ce mémoire se compose de trois parties.

Le premier chapitre reprend succinctement mon curriculum vitae avant de synthétiser mes activités d'Enseignement au sein de l'URCA, aussi bien en termes d'états de service, qu'en termes de Responsabilités pédagogiques. J'y présente également mes activités de Recherche aux travers des encadrements que j'ai réalisés, de ma production scientifique et de ma participation à la vie scientifique au sein et en dehors de mon laboratoire de recherche.

Le deuxième chapitre reprend plus en détails les travaux de recherche que j'ai effectués depuis ma nomination en 2007. Mes activités de recherche ont pour domaine l'Automatique des Systèmes à Événements Discrets (SED) et dépendent du Conseil National des Universités (CNU) 61ème section. Une partie de mes recherches concerne la conception de systèmes sûrs de fonctionnement. J'ai ainsi eu l'occasion d'approfondir des approches de commande et de diagnostic des SED en utilisant des techniques de synthèse ou des outils de vérification formelle. Ces deux thématiques sont développées en deux sous-sections montrant également l'impact que peut avoir la modélisation sur une proposition et l'interaction qu'elles peuvent avoir sur l'Homme (Figure 1). Ces recherches ont été appliquées essentiellement sur les systèmes manufacturiers et de transport.

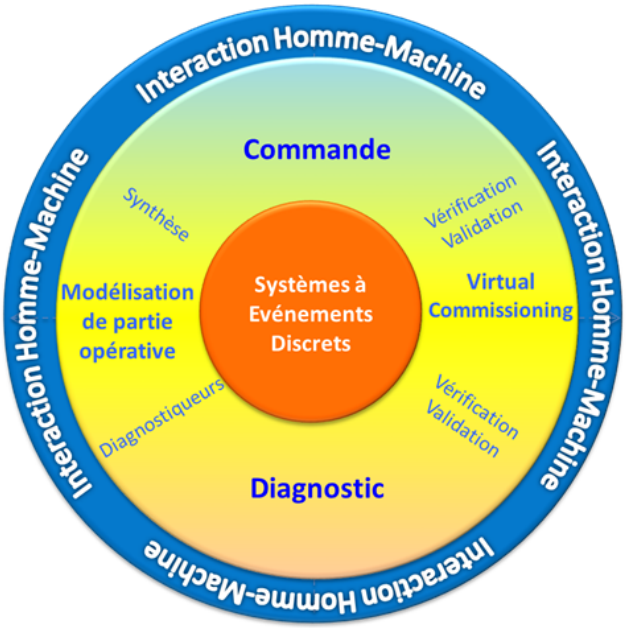

Figure 1 : Domaines d'applications de recherche

Le troisième chapitre reprend des travaux introduisant la composante humaine au sein du système. Elle y présente notamment des résultats obtenus par des doctorants sous contrat CIFRE. Ces travaux ont abouti à un constat sur la place de l'Opérateur Humain au sein de l'industrie. Je présente des pistes de réflexion pour des projets de recherche à court et long termes. Ces perspectives mettent l'Opérateur Humain au premier plan dans l'industrie du futur, que ce soit aussi bien dans un cadre pédagogique qu'industriel (Figure 2). Je développe dans ce chapitre des pistes de recherches permettant :

i) d'aider le concepteur dans l'étape de conception d'une commande au travers d'approches d'Ingénierie des Systèmes à base de modèles (Model-Based SE),

ii) de retourner à l'utilisateur une information de diagnostic qui lui soit fiable et compréhensible (*Situation Awareness & Dissonance*) en prenant en compte les paramètres de dissonances,

iii) d'allier les modules de commande et de diagnostic pour la reconfiguration des systèmes,

iv) d'introduire les problématiques des nouveaux outils de formation que les opérateurs 2.0 auront dans le futur. Ces perspectives sont à la fois des sujets d'actualité dans notre société, dans nos formations et dans mes récentes responsabilités. L'industrie du futur (ou Industrie 4.0) devient un enjeu technologique et sociétal obligatoire qu'il va falloir appréhender au niveau de nos activités de recherche.

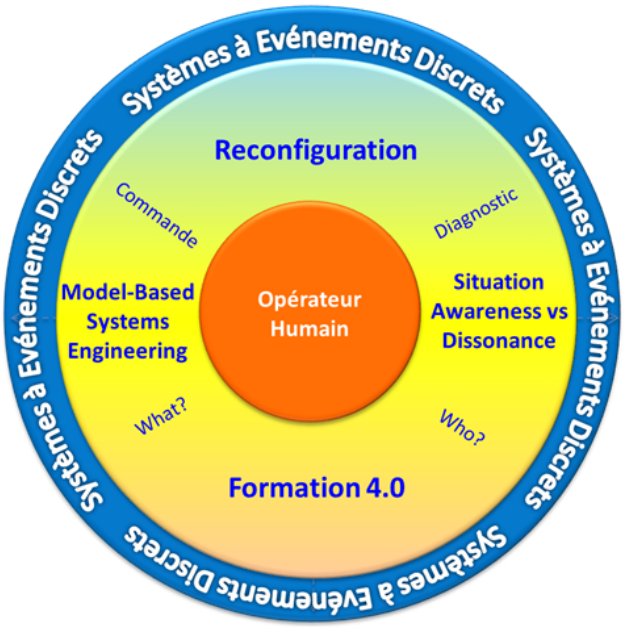

Figure 2 : Perspectives de Recherche

En annexe, le lecteur trouvera deux publications concernant mes travaux.

Chapitre 1 : Synthèse des activités d'enseignement, de recherche et de valorisation

1. Curriculum Vitae

PHILIPPOT Alexandre

Nationalité : Française

Date de naissance : 18/04/1979

Situation familiale : Marié, 2 enfants

Adresse personnelle : 5 rue de Trépail
51400 Les Petites Loges
Tél. : 03.26.48.32.56 **Mobile :** 06.85.01.97.63
Email : aaphilippot@orange.fr

Adresse professionnelle : Centre de Recherche en STIC (CReSTIC) – Bât. 12
UFR Sciences Exactes et Naturelles
Moulin de la Housse – BP 1039
51687 REIMS Cedex 2, FRANCE
Tél. : 03.26.91.86.16
Email : alexandre.philippot@univ-reims.fr

FONCTION ACTUELLE

Maître de Conférences (CNU 61ème section - Génie informatique, automatique et traitement du signal), classe normale à l'Université de Reims Champagne-Ardenne (URCA) depuis le 1er septembre 2007.

Enseignant au département Électronique, Électrotechnique, Automatique (EEA) de l'UFR Sciences Exactes et Naturelles de l'URCA.

Chercheur au Centre de Recherche en Sciences et Technologies de l'Information et de la Communication (CReSTIC) de l'URCA, EA 3804.

FORMATIONS & CURSUS UNIVERSITAIRES

2006 – 2007 : Post-Doctorat au sein du Laboratoire Universitaire de Recherche en Production Automatisée (LURPA) de l'Ecole Normale Supérieure de Cachan.

2002 – 2006 : Doctorat en Génie Informatique, Automatique et Traitement du Signal au Laboratoire d'Automatique et de Microélectronique (CReSTIC-LAM) à l'Université de Reims Champagne-Ardenne (URCA). Directrice de Thèse : Pr. V. Carré-Ménétrier. Soutenue le 18/07/2006. *Mention Très Honorable avec félicitations du jury.*

2001 – 2002 : Diplôme d'Etudes Approfondies (DEA) - Traitement de l'information et Organisation dans les Réseaux, les systèmes Industriels et les systèmes Coopératifs - Optimisation et Sûreté des Systèmes (TORIC/OSS) cohabilitation URCA et Université Technologique de Troyes (UTT).

2000 – 2001 : Maîtrise en Génie Electrique et Informatique Industrielle *avec le titre d'Ingénieur-Maître* de l'IUP Génie Electrique et Informatique Industrielle (GEII) de l'URCA.

1999 – 2000 : Licence en GEII de l'IUP GEII (URCA).

1997 – 1999 : Diplôme Universitaire de Technologie spécialité GEII, option Electronique - IUT de Troyes (10).

1996 – 1997 : Baccalauréat Scientifique - lycée Etienne Oehmichen de Châlons en Champagne (51).

AFFECTATIONS ANTERIEURES & EXPERIENCES PROFESSIONNELLES

2004 – 2006 : Attaché Temporaire d'Enseignement et de Recherche (ATER) en 61ème section à mi-temps au Département EEA de l'URCA, puis au Département du Génie Conditionnement et de l'Emballage (GCE) à l'IUT de Reims.

2002 – 2004 : Enseignant Vacataire pour le Département EEA de l'URCA.

2001 : Stage industriel (6 mois) - ARD-Soliance (Agro-industrie Recherches et Développement) – Programmation et Supervision industrielle (Pomacle - 51).

1999 : Stage industriel de DUT puis CDD de 2 mois - CDTAC Concept (Conception,

Développement, Télégestion, Automatisme, Conformité). Réalisation de cartes électroniques et d'automatismes (Tilloy et Bellay - 51).

CONNAISSANCES TECHNIQUES

Langages pour Automate Programmable Industriel (API), outils de modélisation (GRAFCET, Automates à états, Réseaux de Petri…), Matlab (Simulink, Stateflow), Langage informatique (HTML, C, C++, VB), Bureautique.

LANGUES

Français : langue maternelle. Anglais : lu, écrit et parlé.
Espagnol : notions.

DIVERS

Président/Joueur d'une Association Sportive (AS Mairy sur Marne).

2. Activités d'Enseignement

2.1. Lieux d'enseignement

Mon affectation principale est le département EEA (Electronique, Electrotechnique, Automatique) de l'UFR Sciences Exactes et Naturelles de l'URCA. Ce département se situe sur le campus du Moulin de la Housse et se compose de 26 enseignants permanents dont 25 enseignants-chercheurs, 10 enseignants non-permanents, 2 ingénieurs, 1 secrétaire, et 38 intervenants extérieurs (vacataires, professionnels…).

Par ailleurs, durant mon parcours, j'ai été amené également à intervenir sur 2 sites de l'IUT RCC (Reims – Châlons - Charleville) : (i) sur le site de Châlons-en-Champagne pour y dispenser des Cours, TD, TP au sein du département GIM (Génie Industriel et Maintenance) en 2ème année de DUT et en LP, et (ii) sur le site de Reims pour les départements PEC (Packaging Emballage et Conditionnement – anciennement GCE), INFO (Informatique) et MP (Mesures Physiques) en 1ère, 2ème année ou LP.

Enfin, dans le cadre de la création de la branche A2I (Automatique & Informatique Industrielle) au sein du diplôme d'ingénieur délivré par l'UTT (Université Technologique de Troyes) en convention avec l'URCA, j'interviens également à Troyes sur le site de l'UTT pour un module d'automatisme au premier semestre de branche, avant que les étudiants ne poursuivent leurs études à Reims.

2.2. Enseignements dispensés

Dans le cadre de mes activités d'enseignement, j'ai débuté par des vacations de 2002 à 2004 (117,5 heures éq. TD) pour le département EEA de l'UFR SEN de l'URCA, ainsi que pour l'IUT de Reims-Châlons-Charleville. J'ai par la suite signé un premier contrat de ½ ATER (96h éq. TD) pour l'année universitaire 2004-2005 au sein du département EEA, puis un second pour l'année 2005-2006 avec le département Génie du Conditionnement et de l'Emballage (GCE) de l'IUT de Reims.

Depuis ma nomination en septembre 2007, j'enseigne principalement sur les formations suivantes :

- 3ème année de Licence SPI (Sciences Pour l'Ingénieur) – UFR SEN,
- 1ère et 2ème années de Master EEAII (Electronique, Electrotechnique, Automatique et Informatique Industrielle) – UFR SEN,
- Licence Pro MQ2E (Maîtrise et Qualité de l'Energie Electrique) de l'UFR SEN de Reims,
- 1ère, 2ème et 3ème années de branche Automation & Informatique Industrielle - UTT/URCA,
- 1ère et 2ème années ITII génie mécanique de Champagne-Ardenne - Formation d'ingénieur ENSAM (site de Châlons en Ch.) en convention avec l'URCA.

J'interviens ou j'ai pu intervenir également dans plusieurs formations connexes tels que :

- 1ère et 2ème années DUT PEC (Packaging Emballage et Conditionnement) de l'IUT de Reims,
- 1ère et 2ème années DUT GIM (Génie Industriel et Maintenance) de l'IUT de Reims (site de Châlons en Champagne),
- Licence Pro S3IM (Systèmes d'Information Industriels et Informatique Mobile) de l'IUT de Reims,
- Licence Pro CIM (Capteurs Instrumentation Métrologie) de l'IUT de Reims,
- Licence Pro TAM (Techniques Avancées de la Maintenance) de l'IUT de Reims (site de Châlons en Champagne).

Les disciplines enseignées sont principalement orientées autour des Systèmes à Evénements Discrets, de l'Automatisme, de la Supervision Industrielle, de l'Ingénierie des Systèmes et des Réseaux industriels. Cet enseignement représente une charge moyenne de 282,5 heures/an et est dispensé en moyenne annuelle comme suit Figure 1-3 :

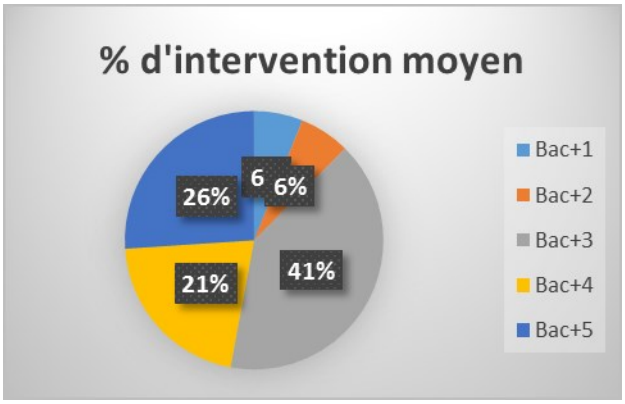

Figure 1-3 : Répartition des enseignements par niveaux

Intervenant principalement dans la spécialité « Systèmes Automatisés » de 2ème année de Master EEAII, j'encadre chaque année plusieurs projets de fin d'études et réalise également le suivi en entreprise d'étudiants lors de leur stage en industrie ou durant leur alternance. Dans une moindre mesure, je réalise la même chose avec des étudiants de la LP MQ2E, de la LP S3IM et du diplôme d'ingénieur en mécanique (ENSAM-URCA-ITII).

Je participe au montage de nouveaux supports de Cours, Travaux Dirigés et surtout de Travaux Pratiques pour les étudiants, notamment en automatisme, communication industrielle et module d'initiation à la Recherche. On retrouve d'ailleurs une partie de ce travail sur des publications dans différents Colloques sur l'Enseignement des Technologies et des Sciences de l'Information et des Systèmes (CETSIS), notamment autour de l'utilisation du *model-checking* [CN13] et la compréhension de la qualimétrie [CN14] de code API. Par ailleurs, j'ai collaboré au 1er partenariat académique entre la société Itris Automation Square et l'URCA en 2012 sur la familiarisation à l'utilisation d'outils de développement de programmes automates.

Les différentes implications et responsabilités que j'ai pu prendre depuis le début de ma carrière, et que je détaillerai ci-dessous, ont été reconnues par l'établissement au travers l'intégration d'heures aux services statutaires (HRS). Ainsi par exemple, les 44 heures HRS en 2017-2018 ont été attribuées au titre de :

- Responsabilité d'option Mécatronique-Robotique (ITII) : 10h
- Responsabilité du Master EEAII (26h cumulées) : 2h
- Charges spécifiques Département EEA : fonctions (30h cumulées) : 3h
- Compensation déplacements Troyes formation A2I : 14h
- Communication (Portes ouvertes - Demi-journée) : 1h
- Suivi de 2 étudiants alternants en 1ère année ITII : 14h

Une table récapitulative est donnée en Tableau 1-1.

Contrat	Année	Bac+1	Bac+2	Bac+3	Bac+4	Bac+5	HRS-PRP-PCA	Total h. Eq. TD	Total h. Eq. TD sans HRS	heures Sup.
Vacataire	2002-2003	13.5	2	15	24	0	0	54.5	54.5	
Vacataire	2003-2004	0	12	25	26	0	0	63	63	
1/2 ATER	2004-2005	0	22	40.66	30	4	0	96.66	96.66	
1/2 ATER	2005-2006	37.33	26.67	32	0	0	0	96	96	
Post Doc	2006-2007	0	0	0	0	0	0	0	0	
MCF	2007-2008	0	0	106	98	35.33	0	239.33	239.33	47.33
	2008-2009	10.67	37.34	124	68	61.67	0	301.68	301.68	109.68
	2009-2010	16	39	176	47	52	0	330	330	138
	2010-2011		46	97	42	68	0	253	253	61
	2011-2012	18	16	136	54	78	10	312	302	110
	2012-2013	18	12	95.75	39	125	6	295.75	289.75	97.75
	2013-2014	40	0	122.5	39	117	20	338.5	318.5	126.5
	2014-2015	12	0	111	39	103	15	280	265	73
	2015-2016	18		118	51	52.5	39	278.5	239.5	47.5
	2016-2017	30		149	65	105.5	40	389.5	349.5	157.5
	2017-2018			84	143	51	44	322	278	86
	2018-2019 Prév.			57	96	71	40	264	224	32
							TOTAL Carrière	**3914.42**	**3700.42**	**1086.26**

Tableau 1-1 : Récapitulatif des heures d'enseignement (Eq. TD)

Ce tableau montre que j'ai toujours dépassé un service statutaire d'enseignement (192h éq. TD), et ce sans que cela soit totalement de ma volonté (Figure 1-4). Si l'on regarde cette évolution dans le graphique de la Figure 1-5, deux pics se détachent fortement. Le premier sur l'année 2009-2010 où j'ai soulagé un collègue afin qu'il puisse rédiger plus sereinement son HDR. Le second pic sur l'année 2016-2017 correspondant à l'ouverture de la branche A2I UTT/URCA et celle de l'option Mécatronique-Robotique du diplôme ITII Mécanique où de nombreuses heures ont dû être assurées que ce soit par mes collègues ou par moi-même. Je reviens actuellement dans une phase respectable d'heures d'enseignement qui contraindront beaucoup moins mes recherches.

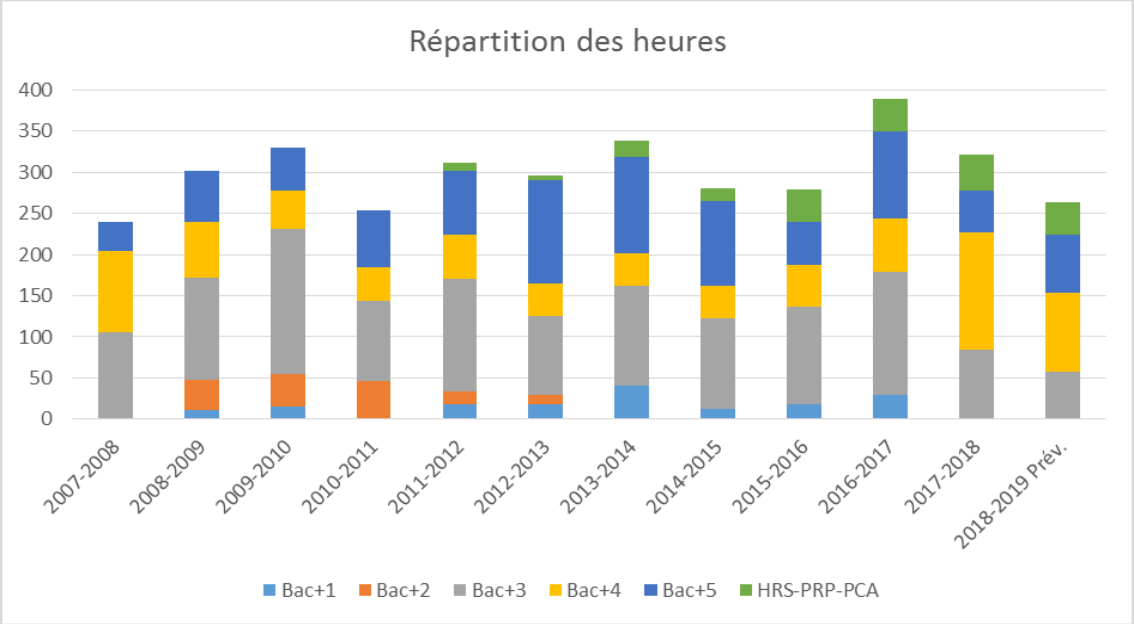

Figure 1-4 : Répartition des heures depuis 2007

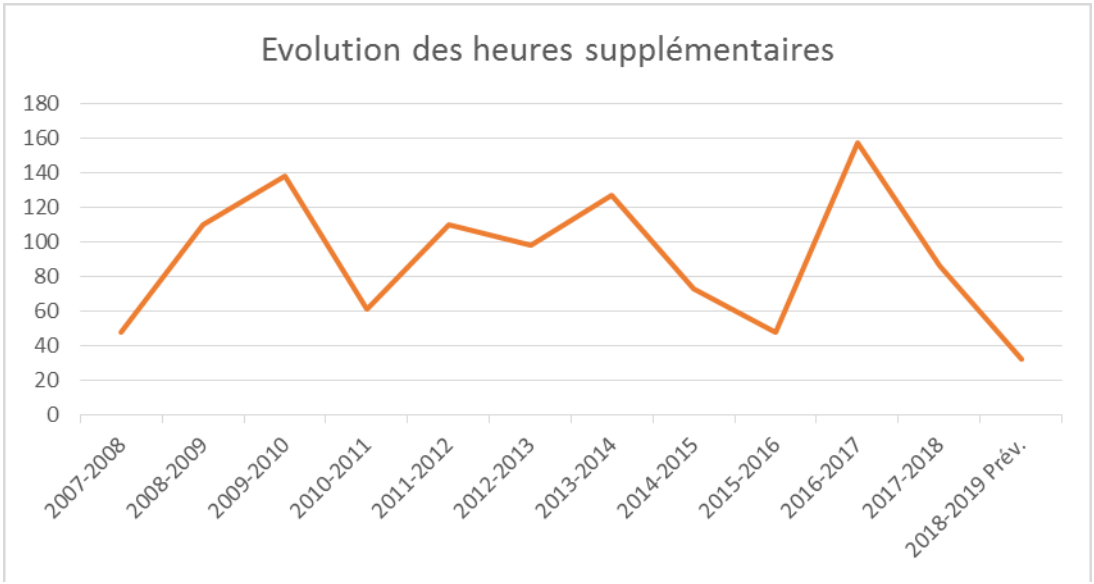

Figure 1-5 : Evolutions des heures supplémentaires

2.3. Disciplines enseignées

Sans détailler l'ensemble de mes enseignements dans chacun des modules auxquels j'interviens, je rappelle brièvement ici certaines disciplines enseignées sur plusieurs niveaux.

Introduction à l'automatisme : Cet enseignement s'adresse aux étudiants Bac+1 dont le cœur de métier n'est pas l'automatisme. Il doit permettre une appréhension de la technologie

au travers l'étude d'une chaîne fonctionnelle d'un Système Automatisé de Production (SAP). Sans programmer un Automate Programme Industriel (API), les étudiants touchent à l'aspect structurel et fonctionnel d'un SAP par une analyse fonctionnelle d'une station de l'atelier flexible CellFlex (pré-actionneur, actionneur, détecteur…). Ils pourront ainsi côtoyer et communiquer avec des automaticiens dans un même langage.

GRAFCET et Automatisme : L'objectif est de donner aux étudiants de Bac+3 à Bac+5 les connaissances nécessaires à la conception et le développement des systèmes automatisés de production. Pour cela, différents outils comme le GRAFCET (définition, concept de base) leur sont présentés pour la spécification de la commande d'un SAP. Par la suite, ils apprennent à programmer un API à travers un langage de programmation de la norme IEC 61131-3 («Automates programmables - Partie 3 : Langages de programmation» International Electrotechnical Commission). Ces modules autour des Systèmes Automatisés de Production sont en perpétuelle évolution avec notamment l'évolution des technologies. Ainsi, les suites logicielles sont régulièrement revues, les parties opératives upgradées. Des supports TP ont par exemple été repris récemment avec notamment l'utilisation des outils HOME I/O et FACTOY I/O (*https://realgames.co/*) comme support de simulations 3D. Cette utilisation fait suite d'un partenariat entre le CReSTIC et la société RealGames depuis 2008.

Automatisation, Supervision et Réseaux de communications industrielles (Bac+3 à Bac+5) : L'objectif est de réaliser une communication entre un ou plusieurs API et un superviseur industriel (TP et TP-Projet). Ces modules doivent permettre aux étudiants d'identifier plusieurs protocoles de communication et mettre en œuvre la chaîne d'informations d'un automatisme. Pour ma part, j'y interviens essentiellement dans la mise en place de clients/serveurs OPC DA avec encore une fois l'utilisation des outils de RealGames : ITS PLC et FACTORY I/O. J'y prépare également un TP autour de OPC UA (Unified Architecture), standard de l'industrie du futur.

Systèmes à Evénements Discrets : Ce module consiste à connaître les principaux outils de modélisation des systèmes à événements discrets (Automates à états, GRAFCET et Réseaux de Petri…). Il vise également à introduire une initiation à la recherche au travers d'applications sur le diagnostic ou la vérification formelle par *model-checking* qui seront poursuivies avec le module suivant.

Méthodologie de conception des systèmes (Bac+4 à Bac+5) : Basé en grande partie sur l'Ingénierie Système et le cycle en V, ce module vise à aborder les méthodes à mettre en œuvre pour insérer rapidement les nouvelles technologies. Les compétences spécifiques visées sont la réalisation des études de conception systèmes, la définition des exigences et des informations techniques nécessaires à la définition du nouveau système, la réalisation des procédures de vérification et validation. C'est également à travers ce module que j'introduis une partie de mes recherches. En effet, en termes d'applications concrètes à la phase de tests et vérification dans la phase remontante du cycle en V, j'enseigne le *model-checking* aux étudiants de Master 2 par des exemples ludiques (exploration exhaustive, problème d'atteignabilité …). C'est notamment dans ce cadre que Mohamed Niang a pu découvrir ce qui allait être son travail de Recherche lors de son doctorat avec la SNCF. Sur une autre formation (ingénieurs A2I), après leur avoir présenté différents outils de modélisation (RdP, automates à états, diagramme SysML …), une étude expérimentale est menée sur la méthodologie de conception à utiliser devant un problème dit simpliste (et pourtant non simple). Cette étude montre un manque flagrant de méthodes de travail dans un chapitre de thèse de Romain Pichard (Pichard, 2018).

Projets & Projets Transversaux : Ces projets se retrouvent évidemment dans beaucoup de nos formations. Très souvent interdisciplinaires, ils se distinguent soit dans le cadre de petits sujets proposés par des industriels ou des enseignants, soit aux travers de concours spécifiques, le tout sur plusieurs semaines d'un semestre. A titre d'exemple, on peut retrouver des sujets de développement de supervision industrielle proposés par Schneider Electric, la participation à des concours d'IS (Robafis) ou de robotique (Olympiades FANUC). Dans ce module, nous faisons très souvent appel à la plateforme CellFlex4.0 qui est un atelier flexible complexe avec un ensemble d'outils de simulation (http://www.univ-reims.fr/meserp/accueil/plateau-de-formation-et-de-transfert-technologique,9485,27019.html).

Par ailleurs, je vais suivre régulièrement des formations technologiques afin de compléter mon apprentissage des technologies et étendre mes champs de compétences :

- Ordinal Software : MES sous Coox
- Formation CBA : Simatic Manager + Imap
- ABB : RobotStudio
- Mitsubishi : Programmation de robots sous Ciros
- Schneider Electric : Variation de vitesse

- B&R : Automatisme et Motion Control
- FANUC : Roboguide, programmation sous Teach Pendant et DCS

2.4. Implications & Responsabilités Pédagogiques

L'implication que l'on a au sein de son département ou de son établissement n'est pas toujours officialisée par un statut. Il n'empêche que chaque collègue qui s'implique dans la vie d'un établissement permet de valoriser celui-ci mais aussi ses propres activités et celles des étudiants. Je suis actuellement :

- Responsable de l'option « Mécatronique-Robotique » de l'ITII génie mécanique de Champagne-Ardenne de l'ENSAM de Châlons en Ch. – depuis sa création en septembre 2016. Option se déroulant à Reims et en partenariat avec le Pôle Formation UIMM (Union des industries et métiers de la métallurgie).
    - A ce titre, à la création de l'option, j'ai établi les fiches d'Unités d'Enseignement (UE) en collaboration avec les différents intervenants. J'établis les emplois du temps, fais le suivi des projets et organise des visites industrielles (10 étudiants max.).
- Responsable du Secteur « Gestion de projet transversal/IS » de la filière d'ingénieurs A2I (Automatique & Informatique Industrielle) de l'UTT en convention avec l'URCA - depuis sa création en septembre 2016.
    - J'effectue le suivi de certains projets et accompagne les étudiants dans leur réalisation. Par exemple, je suis référent enseignant des équipes étudiantes participantes aux concours Robafis et Olympiades FANUC.
- Responsables des unités d'enseignement « Méthodologie de Conception », « *Manufacturing Execution System* » du Master 2 EEAII, du module « Systèmes à événements Discrets » de la Licence 3 SPI de l'URCA.
    - Etablissement des fiches UE et mise en place des supports d'enseignements.
- Co-responsable de la salle Automate Programmable Industriel du département EEA. Salle de maquettes automatisées avec API Schneider Electric, Siemens et GE.
    - Maintenance des maquettes (Mécanique et électrique)
- Co-responsable de l'atelier flexible CellFlex (entité de la plateforme CellFelx4.0) et webmaster du site http://www.univ-reims.fr/meserp.

  - Cette cellule est utilisée pour la découverte de l'automatisme, pour les formations API, réseaux et Interface Homme-Machine. C'est également un support pour les projets étudiants. Elle est par ailleurs utilisée lors des journées portes ouvertes de l'URCA, Classes en FAC, Fêtes de la Science.
- Membre du jury de Licence 3 SPI, Master 1 et Master 2 EEAII depuis 2012.
- Membre du Conseil de Perfectionnement du Master EEAII du département EEA de l'URCA depuis 2014.
- Membre des commissions VAP-VAE du Master EEAII de l'URCA (depuis 2012).

J'ai par ailleurs été auparavant directeur des études du parcours « Système d'Information de Production » (SIP) du Master EEAII spécialité « Systèmes Automatisés » (SA) de 2012 à 2015. Mon rôle consistait à gérer les emplois du temps d'une vingtaine d'étudiants mais également à effectuer le suivi de ceux-ci dans leur recherche de stages industriels. Ce parcours a été ensuite fusionné avec le parcours ISR (Ingénierie des Systèmes Robotisés).

Enfin, je participe également aux évaluations des dossiers « Etudes en France » (anciennement Campus France) pour les formations LP MQ2E, Master 1 et 2 EEAII, ainsi qu'aux recrutements et auditions de l'ITII Mécanique (ENSAM/URCA).

3. Activités de Recherche

Dans cette section, je reprendrai succinctement mes activités de Recherche depuis mon doctorat et les encadrements auxquels j'ai participé ensuite. Le détail de chaque thème sera développé dans le chapitre suivant. A noter que certaines thèses sont confidentielles et ne pourront être totalement approfondies dans ce document.

3.1. DEA, Doctorat & Post Doctorat

Après des études scientifiques dans le département Electronique – Electrotechnique – Automatique de l'UFR SEN de Reims, je me suis orienté vers un Diplôme d'Etudes Approfondies DEA « TORIC : Traitement de l'information et Organisation dans les Réseaux, les systèmes Industriels et les systèmes Coopératifs » en option Optimisation et Sûreté des Systèmes (OSS). Cette filière s'intéresse à la sûreté de fonctionnement des systèmes complexes et à l'optimisation des systèmes de production et de distribution : fiabilité des systèmes, planification et optimisation de la maintenance, détection, localisation et diagnostic

de pannes… Dans ce cadre, j'ai effectué la partie théorique au sein de l'Université Technologique de Troyes (UTT) avant de poursuivre en stage au Laboratoire d'Automatique et de Microélectronique (LAM – aujourd'hui CReSTIC) à l'Université de Reims Champagne-Ardenne. Durant cette partie pratique, j'ai rejoint l'équipe de Recherche Automatique–SED dirigée par Véronique Carré-Ménétrier, Professeure des Universités, dont les travaux portent - sur la modélisation, l'analyse et la synthèse de commande des SED. J'y ai poursuivi sous sa direction un doctorat sur la thématique du diagnostic des Systèmes à Evénements Discrets à base de modèles, nouvelle thématique à l'époque pour le laboratoire appelé Centre de Recherche en STIC (CReSTIC) à partir de 2004.

Titre des travaux : Contribution au diagnostic décentralisé des systèmes à événements discrets : application aux systèmes manufacturiers.

Établissement et discipline : URCA, Génie Informatique, Automatique et Traitement du signal.

Encadrement : Professeure V. Carré-Ménétrier (Directrice de Thèse) & M. Sayed-Mouchaweh (Encadrant).

Date de soutenance : 18 juillet 2006 (Première inscription à l'ED décembre 2002).

Jury de soutenance composé de : Pr. E. Craye (Ecole Centrale de Lille, Rapporteur), Pr. J.J. Lesage (ENS de Cachan, Rapporteur), Pr. M. Combacau (Université de Toulouse III, Examinateur & Président de Jury), Pr. B. Riera (URCA, Examinateur), Pr. V. Carré-Ménétrier (URCA, Directrice de Thèse) et Dr. M. Sayed-Mouchaweh (URCA, Encadrant).

Suite à l'obtention de mon doctorat en 2006, j'ai effectué une année en Post-Doctorat au sein de l'équipe ISA (Ingénierie des Systèmes Automatisés) du LURPA (Laboratoire Universitaire de Recherche en Production Automatisée) de l'ENS de Cachan. Cette année post-doctorale réalisée sous la direction du Professeur Jean-Marc Faure m'a permis notamment de travailler autour de la « Structuration hiérarchique de la commande des systèmes manufacturiers » [CSA3]. Ce travail, en partenariat avec Dassault Systèmes, consistait à évaluer le langage SFC+ (*Sequential Function Chart*) dans l'environnement LCM Studio de l'atelier DELMIA Automation sur des exemples de systèmes manufacturiers.

En 2007, j'ai postulé sur différents postes de Maître de Conférences. Mon choix s'est porté pour un retour à l'URCA, pour des raisons professionnelles, afin de poursuivre mes travaux sur le diagnostic des SED, de développer d'autres travaux en relation directe avec les thématiques de recherche de l'équipe, et pour des raisons personnelles, mon épouse ayant été titularisée dans l'académie de Reims.

3.2. Synthèse des thématiques de Recherche en lien avec mes encadrements

Les thèmes de Recherche développés dans mes travaux portent essentiellement sur la modélisation et l'analyse des SED pour apporter une contribution aux domaines du diagnostic, de la reconfiguration, de la synthèse de la commande et des interfaces homme machine (IHM). Je reprends dans cette section chacun de ces thèmes afin de les mettre en lien avec mes encadrements. Le côté projet sera présenté plus loin.

3.2.1. Modélisation des SED

Un modèle est une représentation abstraite de la réalité afin d'en appréhender plus simplement le sens. Nous construisons des modèles pour les systèmes complexes parce que nous sommes rarement en mesure d'interpréter de tels systèmes dans leur intégralité. En entreprise, les phases de bureau d'études et d'Ingénierie Systèmes (IS) sont alors davantage approfondies. Cependant, chaque étude étant différente, il est important de définir l'objectif à atteindre afin de répondre à la question : « Un modèle : pour quoi faire ? ».

La vitesse, l'accélération, le niveau, la pression, la température, le débit, la tension, le courant sont des variables continues, dans le sens où elles peuvent prendre n'importe quelle valeur lorsque le temps évolue. Les équations différentielles sont l'outil mathématique approprié pour la modélisation, l'analyse et la commande de ces systèmes. Les grandeurs discrètes représentent quant à elles, un nombre de produits dans un stock, un nombre de processeurs en activité, ou bien encore la position d'un produit. L'évolution de ces grandeurs discrètes est conditionnée par l'occurrence d'événements tels que la fin d'exécution d'une tâche, le franchissement d'un seuil, l'arrivée d'un produit, d'un client, la défaillance d'un dispositif décrit un SED comme « un système à espace d'états discret dont les transitions entre états sont associées à l'occurrence d'événements discrets asynchrones ». La succession

d'événements constitue des trajectoires permettant de décrire cet espace d'états. Pour les modèles à événements discrets, l'espace d'états est donc un ensemble discret et l'état change seulement à certains instants du temps, de façon instantanée.

Alors que les systèmes continus s'intéressent à des processus obéissant à des lois physiques, des équations différentielles ou aux dérivées partielles, les SED recouvrent des systèmes ne se souciant que des débuts et des fins de phénomènes (événements discrets) et à leur enchaînement dynamique, logique ou temporel. On retrouve par exemple dans cette catégorie les systèmes manufacturiers, les réseaux de communication, d'énergie et de transport. De nombreux outils permettent la modélisation des SED. La représentation des SED dépend essentiellement de la granularité de modélisation et de l'objectif recherché. Parmi ces outils, on peut citer les automates à états finis, le GRAFCET, les réseaux de Petri…

Un des domaines d'application du laboratoire étant les Systèmes Automatisés de Production (SAP), j'ai cherché à modéliser les composants physiques de ces systèmes complexes à différents niveaux de granularité, et cela afin d'utiliser par la suite ces modèles pour des activités de diagnostic et de synthèse de la commande qui seront décrits par la suite. Les SAP peuvent se décrire en deux parties : une partie opérative (PO) représentant l'équipement, et la partie commande (PC) représentant la partie intelligente du système. La modélisation de la partie opérative consiste à représenter la chaîne fonctionnelle à travers ces éléments que sont : les pré-actionneurs, les actionneurs et les détecteurs. Une description précise du comportement de la PO est souvent une opération complexe car les modèles doivent tenir compte à la fois de la technologie du matériel utilisé, et de l'environnement du procédé. Un des problèmes récurrents à cette modélisation par des outils de représentation des SED est l'explosion combinatoire. Cette explosion correspond à l'incapacité à décrire un comportement par une représentation graphique qui soit interprétable par l'homme et par la machine. Pour contourner ce problème, il convient de « déstructurer » (ou décomposer) le système en sous-ensembles communicants ou non. On parle alors d'approches décentralisées, modulaires, distribuées voire hiérarchiques. Cette démarche doit permettre d'exprimer des causalités simples entre les éléments d'une PO. Mes travaux présentant différents résultats et des études comparatives ont fait l'objet de publications [RN2, CI3, CI11, CI12, CI15, CI17]. Ces modèles ont ensuite été étendus pour la recherche d'approche de diagnostic et commande des SED.

3.2.2. Diagnostic des SED

Maintenabilité et disponibilité des équipements sont les deux mots d'ordre de l'industrie depuis quelques années. Il y a nécessité de connaître l'état des installations afin de prédire et/ou prévenir au plus tôt un comportement défaillant. Les politiques de maintenance évoluent et on parle aujourd'hui de diagnostic et pronostic de systèmes. Une réelle problématique d'observation du comportement des systèmes complexes existe. La plupart des travaux de la littérature autour de cette problématique repose sur l'utilisation de modèles observateurs appelés « diagnostiqueurs » (Sampath, 1994). Cependant, l'obtention de tels modèles nécessite une reconstruction du comportement normal mais aussi anormal du système, ce qui génère un problème d'explosion combinatoire inhérente à tout SED.

Au cours de mon doctorat, j'ai proposé une approche décentralisée permettant d'identifier des composants génériques d'une installation et d'en déduire l'ensemble des défauts pouvant survenir. Cette approche considère la Partie Opérative (PO) comme un ensemble d'éléments composé d'un actionneur et d'un ensemble de capteurs. La construction des diagnostiqueurs s'appuie sur une modélisation modulaire des éléments de la Partie Opérative, d'un modèle des spécifications de la Partie Commande (PC) et d'une information temporelle liée à la réactivité des actionneurs. Chaque diagnostiqueur représente un observateur de l'état du système affecté d'une étiquette de décision, cette décision étant le résultat de l'observation des événements, des conditions sur les états du système et/ou sur le temps de retard entre événements. De là, une recherche de dépendance entre composants est étudiée afin de lever les incertitudes de décisions ou l'apparition de fausses alarmes. Ainsi, plus que la construction de ces diagnostiqueurs, une recherche de la « diagnosticabilité » d'un système a été étudiée (garantir qu'un défaut peut être détecté et isolé avec certitude dans un temps fini).

L'ensemble des décisions locales doit être ensuite agrégé afin d'obtenir une décision globale sur l'état du système. Cette fusion est réalisée par un coordinateur construit à partir d'un ensemble de règles permettant de résoudre les différents problèmes d'indécision et d'ambiguïté entre les diagnostiqueurs locaux. Ce coordinateur permet d'obtenir des performances de diagnostic équivalentes à celles d'un diagnostiqueur centralisé. Les résultats obtenus ont été publiés dans une revue JCR [RI1], 6 conférences internationales [CI5, CI6, CI7, CI8, CI9, CI10] et 2 conférences au niveau national [CN3, CN4].

A la suite de ces premiers travaux, j'ai étendu l'approche sur une structure distribuée de diagnostic afin de totalement ignorer les étapes de composition de modèles, source d'explosion combinatoire. Une aide à la conception de diagnostiqueurs locaux communicants a été proposée [RI2, RN3, CH1, CI13, CI14, CI16, CI17, CI25, CI26, CI27]. Ceci implique alors une redéfinition de la notion de diagnosticabilité pour les approches distribuées. Cette recherche s'appuie sur l'utilisation d'algorithmes de *model-checking* permettant le parcours d'un ensemble de chemins possibles devant vérifier des propriétés spécifiques [RI5, CI31, CI43, CN16]. Une partie de ces travaux a notamment donné lieu à des présentations lors de réunions du Groupe de Travail INCOS (INgénierie de la COmmande et de la Supervision) lors des Journées du Pôle STP du GDR MACS et lors des Journées Doctorales / Journées Nationales MACS [CSA4, CSA5].

Dans la continuité de ces travaux, une thèse a débuté en décembre 2016 avec Mehdi Chankate. Le projet consiste à étendre les études d'évaluation de la diagnosticabilité en proposant une méthode par *model-checking* [CN22]. L'espace mémoire pour cette opération pouvant être importante, l'utilisation du calculateur ROMEO de l'URCA est fortement envisagée.

3.2.3. Synthèse de commande sûre de fonctionnement

L'accroissement de la demande d'automatisation et la complexité des systèmes de production ont conduit à l'utilisation de méthodes formelles. La littérature scientifique propose deux approches pour répondre à cet objectif : i) la théorie de contrôle par supervision (*Supervisory Control Theory* - SCT) (Ramadge et Wonham, 1989) et la Vérification et Validation (V&V) de modèles. Que ce soit l'une ou l'autre des approches, les difficultés de calcul et plus généralement le problème d'explosion combinatoire s'opposent à leur application dans le monde industriel.

J'ai débuté mes recherches sur la synthèse de commande sûre de fonctionnement lors de mon stage de DEA. A l'époque, l'équipe travaillait sur une approche basée sur la SCT avec la définition de contraintes de sécurité et de vivacité appliquées à une commande spécifiée en GRAFCET. Lors de mes travaux, j'ai proposé un algorithme de synthèse permettant d'obtenir non pas un superviseur global du système, mais un sous-ensemble de celui-ci [RN1, CI1, CI4,

CN1, CN2]. Ce sous-ensemble représente une « fenêtre glissante » de la commande la plus permissive possible permettant par construction pas-à-pas d'obtenir l'équivalence d'un superviseur global sans explosion combinatoire.

La recherche sur la synthèse sûre de fonctionnement des contrôleurs logiques a conduit à poursuivre les travaux existants sur la SCT (*Supervisory Control Theory*) dans le cadre de la collaboration avec le Professeur Abdelouhaed Tajer de l'Université Cadi Ayyad à Marrakech (Maroc). Les travaux proposés consistent en la recherche d'une commande dite « optimale » à travers l'utilisation de modèles décentralisés. Les premiers résultats ont permis notamment de diminuer la complexité des modèles utilisés en repoussant l'étape de composition de modèles locaux. En effet, l'approche originale se basant sur la SCT classique demande une étape de modélisation de la partie opérative (PO) du système. Cette étape est très souvent « gourmande » au niveau de la taille des modèles et représente un frein à son applicabilité. La proposition consiste à ne pas disposer d'un modèle global de la PO, mais de plusieurs modèles locaux sur lesquels il est possible d'appliquer des contraintes de sécurité locales. Ces modèles sont ensuite composés afin d'obtenir une commande centralisée dite « raffinée ». Par la suite, et dans un objectif de palier toutes compositions entre modèles source d'explosion combinatoire, une approche par intégration de toutes contraintes (globales ou locales) de sécurité et de vivacité a été proposée. L'idée étant d'obtenir des contrôleurs locaux décentralisés depuis l'approche de SCT localement appliquée [RI3, RI4, CI24, CI28, CI30, CI34, CI40, CN6, CN7, CN17]. Ce travail a été poursuivi avec l'encadrement d'un doctorant au Maroc, Monsieur Yassine Qamsane, sur la conception d'une commande distribuée sur la base des travaux de la SCT. Il a notamment mis en avant de nouvelles règles d'implication pouvant être interprétées dans une spécification GRAFCET. Ces travaux, en plus de conférences internationales, ont donné lieu à 2 revues internationales JCR [RI6, RI7].

Par ailleurs, une méthode originale par filtre à base de fonctions logiques (appelées contraintes), dont les propriétés de sécurité et vivacité sont vérifiées hors ligne par *model-checking*, implémentées à la fin du programme de commande du contrôleur, a été proposée au sein du laboratoire [CN18]. Cette approche originale permet soit de fiabiliser des programmes Automates Programmables Industriels (API) existants, soit de développer des contrôleurs sûrs de fonctionnement originaux où les aspects fonctionnels et sécuritaires sont séparés [CI29, CI32, CI33, CN8, CN9, CSA7, CSA8]. La méthode proposée peut modifier considérablement

la façon de concevoir les programmes API. Cette approche a été testée avec succès sur la cellule CellFlex dans le cadre du projet Mesures des performances et Optimisation des Systèmes de Production (MOSYP) du CPER 2007-2013, et sur les équipements d'alimentation des lignes électrifiées (EALE) de la SNCF dans le cadre de la thèse CIFRE de Raphael Coupat démarrée en 2011 et soutenue en 2014 [CI35, CI37, CI38, CI44, CN12, CN15, CSA11, RN5, RI9]. Elle a fait également l'objet d'une application dans le projet EDUCASCOL (Education cognitive et scolarisation : lutter contre l'échec scolaire en « outillant » les élèves dès les premiers cycles) dans le cadre du CPER 2007-2013 [CI32]. Ce travail a aussi permis de collaborer avec Monsieur David Annebicque durant son Post-Doctorat autour de la conception d'un logiciel interpréteur de GRAFCET [CN10]. Le travail de thèse de R. Coupat a ensuite été étendu dans le cadre d'une seconde thèse CIFRE avec la SNCF sur la vérification formelle des programmes de contrôle/commande. Mohamed Niang a débuté son doctorat en janvier 2016 et a proposé d'utiliser des outils de *model-checking* et de *virtual commissioning* afin d'automatiser un certain nombre de tests de recettes usine pour aider les chargés d'études dans leur travail [CN21, CI54]. Cette thèse devrait être soutenue début janvier 2019.

Ces deux thèmes de Recherche (Diagnostic et Synthèse de la commande) sont d'un point de vue de la chaîne fonctionnelle d'un SAP très liés. Ce rapprochement a notamment été abordé dans un projet du Groupement d'Intérêt Scientifique « Surveillance, Sûreté, Sécurité des Grands Systèmes » en 2011. Ce projet nommé ADEXEC (Approche de Détection et d'EXplication d'Erreur de Commande par filtrage robuste) a fait l'objet d'un partenariat avec une équipe du Centre de Recherche en Automatique de Nancy (CRAN) et du Laboratoire d'Automatique, de Mécanique et d'Informatique Industrielles et Humaines (LAMIH) de Valenciennes. Ce projet s'intéressait aux systèmes critiques, pilotés par des opérateurs humains, qui nécessitent une aide à la conduite et avait pour objectifs de proposer une approche assurant la sécurité des personnes et de l'environnement, ainsi que le « bon » pilotage du système en modes normal et dégradé. Une approche de surveillance utilisant à la fois l'approche de diagnostiqueur et le filtre de commande a ainsi permis d'obtenir une commande tolérante aux fautes [RN4, CI36, CI39, CSA6].

Enfin, le groupe AUTO (devenu aujourd'hui département ATS) a obtenu en 2015 une allocation doctorale dans le cadre de l'appel à projet Essaimage sur le diagnostic et la

commande par contraintes logiques. Ce travail vise à établir un lien entre les approches de commande sûre de fonctionnement des SED à base de contraintes logiques et les approches de diagnostic de SED à base de modèles. Le doctorant Romain Pichard a notamment formalisé l'ensemble du filtre logique et a établi des conditions de cohérence et de suffisance pour l'ensemble des contraintes [CI53, CI57, RI8]. Cette thèse devrait être soutenue le 30 novembre 2018.

3.2.4. Coopération Homme-Machine

Dans le cadre de mes activités de modélisation, j'ai co-encadré la thèse de Mathieu Hemour, en convention CIFRE avec la RATP (novembre 2007-décembre 2010) sous la direction du Professeur Bernard Riera, co-encadrée par Nadhir Messaï du CReSTIC et Didier Caligny de la RATP. Son travail intitulé « Contribution à la modélisation et à la simulation appliquées aux études d'exploitation du réseau ferré urbain », a permis la modélisation du réseau ferroviaire urbain afin d'en extraire une série de simulation en vue du dimensionnement des installations mais également de rejouer des scénarios de journées passées [CI19, CI21, CN5]. Ce travail consistait à modéliser essentiellement l'équipement. Cependant, les données issues d'un système peuvent permettre sa modélisation mais également parfois d'identifier un comportement humain s'y appliquant. C'est donc suite aux résultats obtenus qu'une seconde thèse en convention CIFRE avec la RATP a pu débuter en novembre 2014. Cette thèse, intitulée « Compréhension fine du comportement des lignes des réseaux métro, RER et tramway », a été réalisée par Vincent Dimanche sous la direction du Professeur Bernard Riera, co-encadrée par Alban Goupil du CReSTIC et Alain Urban pour la RATP. Ce travail consistait à modéliser un comportement humain, celui des conducteurs, afin de déterminer leur degré de vigilance (hypo ou hyper) selon les créneaux horaires (heures pleines ou creuses), la ligne d'affectation (Métro, RER, Tram) et le type de matériel roulant. Par ailleurs, un travail sur le *Visual Analytic* a été réalisé afin de retourner aux exploitants des indicateurs visuels leur permettant une prise de conscience de la situation (Situation Awareness). Malgré la confidentialité, quelques résultats ont pu être publiés à l'international [CI45, CI52]. Le candidat a soutenu en juin 2018 et V. Dimanche est devenu chargé d'études en sécurité ferroviaire à la RATP.

3.2.5. Reconfiguration des SED

L'établissement d'une commande d'un SAP repose sur des connaissances structurelles du système mais aussi sur ses aspects fonctionnels. L'établissement d'approches de diagnostic permet notamment de détecter les comportements aux dérives anormales mais aussi d'en localiser les causes. Cependant, l'étape décisionnelle suite au diagnostic d'un défaut repose soit sur la considération de ce défaut tout en continuant de produire, on parle alors de Commande Tolérante aux Fautes (*Fault Tolerant Control* - FTC), soit sur la reconfiguration du processus de commande ou de l'atelier. Il s'agit alors de Reconfiguration (commande ou process). C'est dans ce cadre que la thèse d'Imane Tahiri en cotutelle (direction V. Carré-Ménétrier et A. Tajer) a pu débuter en 2017. C'est un thème nouveau pour l'équipe CDSED du laboratoire CReSTIC et les premiers résultats ont d'ores et déjà permis de publier dans des conférences nationales [CN19] et internationale avec notamment un *Best Student Paper Award* [CI55].

3.3. Encadrement de Recherche

Depuis ma nomination, j'ai eu l'occasion de participer à l'encadrement de recherche autour des différents thèmes de la section précédente. Pour chaque thèse soutenue ou en cours, je synthétise dans ce paragraphe le déroulement et y indique ma participation (Tableau 1-2). Un résumé est également associé. Je termine cette section par les encadrements d'étudiants Master Recherche et Ingénieurs.

	Début	**Durée**	**Thématique**	**Financement**	**Taux**	**Production**
M. Hemour	11/07	37 mois	OH	CIFRE	33%	2 CI, 1 CN
N. Malki	10/08	57 mois	Diag	Région	50% sur 6 mois	4 CI
R. Coupat	11/11	36 mois	Cmde, OH	CIFRE	33%	1 RI, 5 CI, 3 CN, 1 CSA
Y. Qamsane	09/14	43 mois	Cmde	Sans (Maroc)	50%	2 RI, 5 CI, 2 CN
V. Dimanche	04/15	38 mois	OH	CIFRE	33%	2 CI
R. Pichard	10/15	38 mois	Cmde	Région	33%	1 RI, 3 CI, 1 CN, 1 CSA
M. Niang	01/16	35 mois	Cmde, OH	CIFRE	33%	1 RI, 1 CI, 1 CN, 1 CSA
M. Chankate	12/16	En cours	Diag	Employeur	33%	1 CN
I. Tahiri	10/17	En cours	Cmde, Diag	Cotutelle	33%	1 CI, 1 CN

Tableau 1-2 : Résumé des encadrements

3.3.1. Encadrements de Thèses Soutenues

- Mathieu HEMOUR
  Titre des travaux : *Contribution à la modélisation et à la simulation appliquées aux études d'exploitation du réseau ferré urbain*.
  Établissement et discipline : URCA, Génie Informatique, Automatique et Traitement du signal.
  Financement : CIFRE RATP
  Encadrement : B. Riera (Directeur de Thèse) 33%, A. Philippot (Co-encadrant) 33%, N. Messaï (Co-encadrant) 33%.
  Première inscription à l'ED : novembre 2007.
  Date de soutenance : 15 décembre 2010.
  Jury : Pr. A. Bernard (Ecole Centrale de Nantes, Président/Examinateur), Pr. W. Schön (Université Technologique de Compiègne, Rapporteur), Pr. F. Vanderhaegen (Université de Valenciennes et du Hainaut-Cambrésis, Rapporteur), Pr. J.J. Lesage (Ecole Normale Supérieure de Cachan, Examinateur), J. Martres (Directeur du département Maîtrise d'Ouvrage du Transport – RATP), D. Caligny (Responsable Domaine Métro - département MOT – RATP, Encadrant industriel), Dr. N. Messaï (URCA, Co-encadrant), Dr. A. Philippot (URCA, Co-encadrant), Pr. B. Riera (URCA, Directeur).
  Production scientifique : 2 communications internationales [CI19, CI21], 1 communication nationale [CN5].
  Devenir du candidat : Ingénieur Département MOP - RATP, Paris.
  Remarque : Le contrat CIFRE avec la RATP n'a pas permis de publier l'ensemble des travaux pour cause de confidentialité totale.

Résumé :

L'exploitation ferroviaire du réseau francilien est en pleine évolution. D'une part, avec l'obsolescence de ses équipements, de nouveaux systèmes de conduite et de régulation sont progressivement mis en place. D'autre part, avec la montée croissante du trafic de ces dernières années, les marges d'exploitation qui permettaient d'absorber les micros perturbations se retrouvent réduites de manière significative. La façon de penser et de définir les études qui mettent en place l'exploitation est donc impactée. Ce mémoire de thèse présente une approche qui vise à améliorer les études d'exploitation ferroviaire en prenant en compte les problématiques actuelles. Tout d'abord, à la suite d'une phase d'expression fonctionnelle menée au sein du département Maîtrise d'Ouvrage du Transport (MOT) de la Régie Autonome des Transports Parisiens (RATP), les différents besoins de modélisation et de simulation ont été regroupés. Puis, après une étude fonctionnelle des outils internes et externes à la RATP, un nouvel outil de simulation dynamique du trafic ferroviaire a été acquis. En s'appuyant notamment sur les possibilités de modélisation et de simulation

apportées par cet outil, de nouvelles méthodologies d'études ont été développées afin de prendre en compte les nouveaux enjeux de l'exploitation ferroviaire et ce dans deux domaines : le dimensionnement de l'infrastructure et l'analyse du trafic. Un exemple d'application à la définition d'un nouveau réseau de transport combiné à l'étude de sa stabilité et de sa robustesse face à des perturbations types illustre les méthodologies mises en place et montre l'intérêt d'une étude complète afin de démontrer l'exploitabilité d'un projet de transport [CI19, CI21, CN5].

- Noureddine MALKI
  Titre des travaux : *Contribution au Diagnostic des Systèmes à Evènements Discrets par modèles temporels et distributions de probabilité*.
  Établissement et discipline : URCA, Génie Informatique, Automatique et Traitement du signal.
  Financement : Bourse Régionale
  Encadrement : N. Manamanni (Directeur de Thèse) 50%, A. Philippot (Encadrant) 50%.
  Première inscription à l'ED : octobre 2008.
  Date de soutenance : 15 juillet 2013.
  Jury : Pr. J.-F. Pétin (Université de Lorraine, Président/Examinateur), Pr. P. Berruet (Université de Bretagne Sud, Rapporteur), Pr. S. Debernard (Université de Valenciennes et du Hainaut-Cambrésis, Rapporteur), Pr. B. Riera (URCA, Examinateur), Dr. A. Philippot (URCA, Encadrant), Pr. N. Manamanni (URCA, Directeur).
  Production scientifique : 3 communications internationales (avant 2011) et 1 communication internationale commune [CI18]
  Devenir du candidat : Ingénieur consultant de Manpower pour Alstom (Grenoble) jusqu'en 2015. Actuellement Ingénieur consultant de Sogeti.
  Remarque : M. Malki a débuté son doctorat en 2008 sous une autre direction qui a été interrompue pendant une longue période. En accord avec l'école doctorale, c'est en décembre 2012 que M. Manamanni et moi-même avons repris la suite et la fin de l'encadrement de la thèse. Ceci explique notamment le peu de publications communes avec le doctorant.

Résumé :

Ce mémoire de thèse présente une contribution au problème de diagnostic des Systèmes à Evénements Discrets (SED). Une démarche de diagnostic en exploitant l'aspect temporel caractérisant l'occurrence des événements est proposée. Le système est modélisé par des graphes temporels appartenant au formalisme des automates temporisés. L'approche est conçue selon une architecture décentralisée afin d'éviter toute explosion combinatoire dans la construction des modèles. Elle a permis la détection et localisation des défauts abrupts

survenant sur les équipements notamment en combinant des conditions d'autorisation d'événements et des fonctions de non-occurrence d'événements. Les défauts graduels issus du process sont également considérés. Pour cela, les contraintes temporelles exprimant les dates d'occurrence des événements dans les *Templates* et les Chroniques sont modélisées par des distributions de probabilités (DP). Celles-ci sont utilisées afin de caractériser un fonctionnement normal, dégradé ou défaillant de chaque sous-système avec un certain degré de certitude. Cette identification du fonctionnement est représentée par la valeur d'un indicateur de dégradation.

- Raphaël COUPAT
  Titre des travaux : *Méthodologie pour les études d'automatisation et la génération automatique de programmes Automates Programmables Industriels sûrs de fonctionnement. Application aux Equipements d'Alimentation des Lignes Electrifiées.*
  Établissement et discipline : URCA, Génie Informatique, Automatique et Traitement du signal.
  Financement : CIFRE SNCF
  Encadrement : B. Riera (Directeur de Thèse) 33%, A. Philippot (Co-encadrant) 33%, D. Annebicque (Co-encadrant) 33%.
  Première inscription à l'ED : novembre 2011.
  Date de soutenance : 27 novembre 2014.
  Jury : Pr. S. Lahaye (Université d'Angers, Président/Examinateur), Pr. A. Toguyeni (Ecole Centrale de Lille, Rapporteur), Dr-HDR A. Subias (Université de Toulouse, Rapporteur), M.-A. Buret (Ingénieur SNCF, Encadrant industriel), Dr. A. Annebicque (URCA, Co-encadrant), Dr. A. Philippot (URCA, Co-encadrant), Pr. B. Riera (URCA, Directeur).
  Production scientifique : 1 revue internationale [RI9], 5 communications internationales [CI35, CI37, CI38, CI44, CI47], 3 communications nationales [CN12, CN14, CN15] et 1 communication sans acte dans une session poster/démonstrateur [CSA12]
  Devenir du candidat : Chargé d'études Automatismes CES 2 – Conception et Expertise EALE, SNCF – INFRA, La Plaine Saint Denis.

Résumé :

Le projet de recherche présenté dans cette thèse a été réalisé avec la collaboration de la Direction de l'Ingénierie SNCF et le CReSTIC de l'Université de Reims Champagne-Ardenne (URCA). L'objectif de ce projet est de contribuer à l'amélioration des études de conception du contrôle/commande des projets d'électrification menées par les chargés d'études. Ce projet doit répondre à des objectifs humains, économiques et techniques exprimés par la SNCF, notamment appliqué au domaine des Equipements d'Alimentation des Lignes Electrifiées (EALE). Pour répondre à ces problématiques, une méthodologie pour les études

d'automatisation est proposée. Elle intègre deux axes de recherche. Le premier axe est la génération automatique de livrables (codes, documents, schémas…). Celle-ci repose nécessairement sur une standardisation et une modélisation du « métier ». L'approche MDD (*Model Driven Development*) du génie logiciel et l'approche DSM (*Domain Specific Modeling*), apportent des éléments de solution reposant sur l'utilisation de *templates* métiers. Toutefois, il est fondamental de générer des livrables de qualité et du code API (Automates Programmables Industriels) sûr de fonctionnement. Le second axe de recherche s'intéresse à la commande sûre de fonctionnement. Trois approches de synthèse de la commande (*Supervisory Control Theory*, synthèse algébrique, commande par contraintes logiques) permettant a priori de répondre à ces objectifs de sûreté sont présentées et discutées. La commande par contraintes logiques présente l'avantage majeur de séparer la sécurité (qui est vérifiée formellement hors ligne par *model-checking*) et le fonctionnel, et de pouvoir être utilisée avec des programmes API existants, ne remettant ainsi pas en cause la méthodologie de travail des chargés d'études.

- Yassine QAMSANE
  Titre des travaux : *Synthèse et Implémentation de la Commande Distribuée des Systèmes à Evénements Discrets : Application aux Systèmes Manufacturiers de Production*.
  Établissement et discipline : Cadi Ayyad University, Marrakech (Maroc), Génie Informatique, Automatique et Traitement du signal.
  Financement : Sans (Thèse au Maroc)
  Encadrement : A. Tajer (Directeur de Thèse) 50%, A. Philippot (Encadrant) 50%.
  Première inscription à l'ED du Maroc : septembre 2014
  Date de soutenance : 22 mars 2018
  Jury : Pr. V. Carré-Ménétrier (URCA, Présidente/Rapporteur), Pr. A. Haqiq (FST Settat, Rapporteur), Pr. M. Nemiche (FS Agadir, Rapporteur), Dr. A. Nait-Sidi-Moh (UPJV, Examinateur), Pr. S. Belkouch (ENSA Marrakech, Examinateur), Pr. E.H. Chakir El Alaoui (FSTG Marrakech, Examinateur), Dr. A. Philippot (URCA, Co-encadrant), Pr. A. Tajer (ENSA Marrakech, Directeur).
  Production scientifique : 2 revues internationales [RI7, RI8], 5 communications internationales [CI40, CI46, CI48, CI49, CI51] dont le *Best Student Paper Award* pour [CI48], 2 communications nationales [CN17, CN19].
  Devenir du candidat : Post-Doctorat à l'Université du Michigan (équipe de D. Tilbury).

Résumé :

Le travail présenté dans ce mémoire de thèse apporte sa contribution à la synthèse de la

commande distribuée des Systèmes à Evénements Discrets (SED) et plus particulièrement pour les systèmes manufacturiers de production. L'approche consiste à proposer une synthèse locale, puis une synthèse globale. Pour cela, elle considère la Partie Opérative (PO) comme un ensemble de sous-systèmes constitués autour d'un actionneur discret lié à un ensemble de capteurs discrets. La synthèse locale consiste à construire des Contrôleurs Locaux (LC) en s'appuyant sur une modélisation modulaire par automates des Eléments de Partie Opérative (EPO) et une modélisation sous forme d'expressions booléennes des spécifications de la Partie Commande (PC). Chaque CL commande ainsi un sous-système individuel en se basant sur une observation locale des événements de celui-ci. Ensuite, pour atteindre l'objectif global, l'information globale est rajoutée aux CL pour élaborer des Contrôleurs Distribués (CD) permettant d'exécuter les actions de commande de manière coopérative. Les CD sont des modèles abstraits, simples, adaptables aux redéfinitions des cahiers des charges, et réduisent le problème de complexité de calcul. L'approche se distingue en considérant des automates à états avec des expressions booléennes qui mènent à une réduction significative de la complexité du calcul. En effet, elle propose non seulement une méthode de synthèse de la commande utilisant des formalismes particulièrement adaptés mais aussi une technique de vérification formelle de la commande obtenue en se basant sur la technique du *model-checking*. De plus, notre approche offre la possibilité d'interpréter la commande obtenue en un langage normalisé de spécification des systèmes de commande pour des fins d'implémentation dans un API. Un exemple basé sur un processus manufacturier réel de taille importante illustre les apports de l'approche.

- Vincent DIMANCHE
  Titre des travaux : *Compréhension fine du comportement des lignes des réseaux métro, RER et tramway. Réalisation des études d'exploitabilité & Définition de nouveaux algorithmes de régulation*.
  Établissement et discipline : URCA, Génie Informatique, Automatique et Traitement du signal.
  Financement : CIFRE RATP
  Encadrement : B. Riera (Directeur de Thèse) 33%, A. Philippot (Co-encadrant) 33%, A. Goupil (Co-encadrant) 33%.
  Première inscription à l'ED : 1er Avril 2015
  Date de soutenance : 11 juin 2018.
  Jury : Pr. S. Debernard (Université de Valenciennes et du Hainaut-Cambrésis, Président/Examinateur), Dr. S. Sadeghian (RATP, Examinateur), Dr. A. Keziou (URCA, Examinateur), Pr. A. Bernard (Ecole Centrale de Nantes, Rapporteur), Dr. L. Oukhellou (Directrice de Recherche GRETTIA – IFSTTAR, Rapporteur), Dr. A.

Goupil (URCA, Co-encadrant), Dr. A. Philippot (URCA, Co-encadrant), Pr. B. Riera (URCA, Directeur), Y. Kaminagai (Invité, RATP), A. Urban (Invité, RATP).
Production scientifique : 2 communications internationales [CI45, CI52].
Devenir du candidat : Chargé d'études sécurité ferroviaire RATP.
Remarque : Cette thèse représente la continuité de la thèse de Monsieur M. Hemour soutenue en décembre 2010 dont je suis également co-encadrant. Elle fait notamment l'objet d'une confidentialité totale d'où le peu de publication (en cours d'une RI).

Résumé :

Les réseaux ferroviaires en milieu dense font face à des saturations importantes. Par ailleurs, l'adéquation entre l'offre théorique et la demande croissante impose des contraintes d'exploitabilités fortes. Ce déséquilibre génère des points conflictuels comme des goulets d'étranglement avec pour effet des retards sur les trains en amont. Comme le facteur humain, parmi une multitude, influence l'exploitation ; le prendre en compte plus finement devrait améliorer la compréhension et la modélisation des lignes pour en accroître la capacité sans sacrifier le confort des passagers. Pour répondre à cet objectif, nos travaux reposent sur une visualisation adaptée des données remontées de l'exploitation et sur leur fouille automatisée. Elles ont été adaptées et appliquées au domaine ferroviaire notamment aux lignes des réseaux ferrés exploités par la RATP. Le processus *Visual Analytics*, mis en œuvre dans nos travaux pour répondre à ces besoins, englobe les étapes nécessaires à la valorisation de la donnée, allant de leur préparation à l'analyse experte en passant par leur représentation graphique et par l'utilisation d'algorithmes de fouilles de données. Parmi ces derniers, le *CorEx* et le *Sieve* nous ont permis par un apprentissage non supervisé basé sur une mesure de l'information mutuelle multivariée d'analyser les données d'exploitation pour en extraire des caractéristiques du comportement humain. Enfin, nous proposons aussi une visualisation intuitive d'une grande quantité de données permettant leur intégration et facilitant le diagnostic global du comportement des lignes ferroviaires.

- Romain PICHARD
  Titre des travaux : *Contribution à la Commande des Systèmes à Événements Discrets par Filtre Logique.*
  Établissement et discipline : URCA, Génie Informatique, Automatique et Traitement du signal.
  Financement : Bourse Régionale – Programme Essaimage
  Encadrement : B. Riera (Directeur de Thèse) 33%, A. Philippot (Co-encadrant) 33%, R. Saddem (Co-encadrant) 33%.
  Première inscription à l'ED : 1er octobre 2015.
  Date de soutenance : 30 novembre 2018.

Jury : Pr. J.F. Pétin (Université de Lorraine, Examinateur), Pr. V. Carré-Ménétrier (URCA, Examinatrice), Pr. Serge Debernard (Université Polytechnique des Hauts-de-France, Examinateur) , Dr-HDR J.M. Roussel (ENS Paris-Saclay, Rapporteur), Dr-HDR L. Piétrac (INSA de Lyon, Rapporteur), Dr. P. Marangé (Université de Lorraine, Invitée), Dr. R. Saddem (URCA, Co-encadrant), Dr. A. Philippot (URCA, Co-encadrant), Pr. B. Riera (URCA, Directeur).
Production scientifique : 1 revue internationale [RI8], 3 communications internationales [CI50, CI53, CI57], 1 communication nationale [CN20] et 1 communication sans acte dans une session poster/démonstrateur [CSA14].
Devenir du candidat : Ingénieur en automatisme et informatique industrielle chez SASP Services.

Résumé :

Cette thèse contribue à une approche formelle de conception d'un programme de contrôle/commande pour les systèmes automatisés de production (SAP) contrôlés par des automates programmables industriels (API). Dans ce contexte, deux constats principaux ont été soulevés : il existe un manque de méthodologie efficace pour la conception d'un programme API dans le monde industriel et les méthodes formelles issues du monde académique ne sont ni connues ni utilisées par l'industrie car trop complexes. Par ailleurs, l'industrie du futur nécessitera des contrôleurs toujours plus flexibles et fiables. La flexibilité implique que les programmes seront encore plus difficiles à réaliser, et par conséquent, la difficulté pour garantir la fiabilité de ceux-ci sera accrue. Pour répondre à ces problématiques, une méthode de conception formelle s'intégrant dans un cycle de développement industriel classique (cycle en V) a été proposée. De plus, afin de faciliter le transfert vers l'industrie tant d'un point de vue technique (API) qu'humain (pratique des automaticiens), le formalisme utilisé est entièrement basé sur des variables et des équations logiques appelées contraintes logiques. Ces contraintes logiques permettent la spécification des exigences informelles recensées dans le cahier des charges. A partir de ces contraintes logiques, un algorithme de résolution des contraintes, implémentable dans un API, est synthétisé et implémenté automatiquement dans un langage de programmation normalisé pour API. Ce filtre logique peut être utilisé pour : commander un SAP contrôlé par un API, vérifier formellement un programme API, mettre en sécurité un programme API déjà existant présentant des erreurs. Les travaux de cette thèse ont eu pour objectif de lever certains verrous et de globalement améliorer et renforcer l'approche par filtre logique. Dans le but de généraliser l'approche par filtre, un effort important a été réalisé autour de la formalisation des contraintes logiques, des différentes fonctions, et propriétés associées au filtre logique. Cet apport de formalisation a

permis, en particulier, de proposer une approche de vérification formelle de la notion de cohérence d'un filtre logique ainsi qu'une condition nécessaire et suffisante à cette propriété. Enfin, après avoir mis à jour l'algorithme d'implémentation classique, deux algorithmes de recherche locale d'une solution basés sur des techniques de solveur SAT ont été proposés.

- Mohamed NIANG
  Titre des travaux : *Vérification formelle et simulation pour la validation des programmes API des EALE (Equipements d'Alimentation des Lignes Electrifiées)*.
  Établissement et discipline : URCA, Génie Informatique, Automatique et Traitement du signal.
  Financement : CIFRE SNCF.
  Encadrement : B. Riera (Directeur de Thèse) 33%, A. Philippot (Co-encadrant) 33%, F. Gellot (Co-encadrant) 33%.
  Première inscription à l'ED : 28 janvier 2016.
  Date de soutenance : 20 décembre 2018.
  Jury : Pr. S. Charbonnier (INP Grenoble, Examinatrice), Pr. D. Trentesaux (UPHF, Examinateur), Pr. S. Lahaye (Université d'Angers, Rapporteur), Pr. J.J. Lesage (ENS Paris-Saclay, Rapporteur), Dr. R. Coupat (Ingénieur SNCF, Encadrant industriel), Dr. F. Gellot (URCA, Co-encadrant), Dr. A. Philippot (URCA, Co-encadrant), Pr. B. Riera (URCA, Directeur), S. Lefebvre (Ingénieur SNCF, Invité).
  Production scientifique : 1 revue internationale [RI9] avec une contribution secondaire par rapport à R. Coupat, 1 communication internationale [CI54], 1 communication nationale [CN21] et une communication sans acte de type Poster [CSA12].
  Remarque : Cette thèse représente la continuité de la thèse de Monsieur M. Coupat soutenue en novembre 2014 dont je fus également co-encadrant (également en confidentialité partielle). M. Niang fût également un étudiant de Master 2 à qui j'ai pu faire découvrir le *model-checking*.
  Devenir du candidat : Chargé d'études Automatismes CES 2 – Conception et Expertise EALE, SNCF – INFRA, La Plaine Saint Denis.

Résumé :

Cette thèse a pour but de mettre en place des méthodes permettant aux chargés d'études de la SNCF de pouvoir vérifier automatiquement les programmes Automates Programmables Industriels (API) et le câblage des armoires électriques. L'approche utilisée dans le 1er axe est basée sur les méthodes formelles de vérification. Elle permet non seulement de vérifier les programmes API en s'inspirant de la méthode actuellement utilisée par les chargés d'études de la SNCF (exécution des instructions du cahier de recettes sur les programmes pour vérifier que ceux-ci respectent les spécifications fonctionnelles et sécuritaires), mais aussi de les vérifier de manière exhaustive. A travers cette dernière approche, il a été prouvé que ces programmes n'étaient pas formellement sûrs de fonctionnement. Une étude sur la cohérence

des contraintes de sécurité du filtre logique de commande par contraintes a également été travaillée (via le *model-checking*). Ce filtre, dont l'objectif est d'interdire toute commande dangereuse pour le système, est implémenté dans les programmes API de la SNCF pour garantir leur sûreté. Pour l'axe 2, un cahier des charges d'un banc d'essai conforme aux besoins des chargés d'études a été établi en vue de son implémentation et évaluation.

3.3.2. Encadrements de Thèses en cours

- Imane TAHIRI
  Titre des travaux : *Contribution à la reconfiguration de la commande distribuée des systèmes à événements discrets.*
  Établissement et discipline : URCA, Génie Informatique, Automatique et Traitement du signal.
  Financement : Cotutelle France/Maroc
  Encadrement : V. Carré-Ménétrier (Co-Directrice de Thèse) 33%, A. Tajer (Co-Directeur Maroc) 33% et A. Philippot (Co-encadrant) 33%.
  Première inscription à l'ED (France) : 1er octobre 2017 (à l'ED Maroc octobre 2016)
  Date de soutenance : prévue en décembre 2019
  Production scientifique : 1 communication internationale avec le *Best Student Paper Award* [CI55] et 1 communication nationale [CN19].
  Remarque : Cette thèse fait suite aux contacts obtenus après ma mobilité sortante en octobre 2015.

Résumé :

De nos jours, il ne convient plus seulement de détecter et de localiser une défaillance dans un système manufacturier, mais également de fournir une solution permettant à l'opérateur d'interagir avec son système. Afin de répondre aux problèmes de sécurité opérationnelle dans le domaine du contrôle des systèmes, la mise en œuvre des méthodes formelles est nécessaire. Dans ce contexte, il est important de surveiller le système et de proposer une solution alternative pour maintenir la production. Pour cela, une reconfiguration du contrôleur est demandée. Dans cet objectif, ce travail de thèse présente une méthode de reconfiguration dans le cas de la détection d'un défaut capteur. L'idée principale de l'approche proposée est d'assurer la continuité des services du système si un défaut capteur se produit et entrave son comportement normal. La continuité des travaux inclut la vérification du contrôle de synthèse temporisée proposée pour le mode défectueux ou reconfiguré. En effet, l'ajout de ces modèles doit permettre de renforcer la sûreté de fonctionnement en mode dégradé à l'image des approches de FTC (*Fault Tolerant Control*) mais implique de vérifier que d'autres

situations/combinaisons n'empêchent pas le fonctionnement du système ou ne le bloquent pas. Par ailleurs, il est nécessaire de développer l'axe de reconfiguration des SED en cas de changement de la configuration du système ou en cas de changement des spécifications suite à une demande de l'opérateur. Une information de retour pour l'opérateur sur la réparation du capteur défectueux peut être aussi prise en compte. Par conséquent, cette information permettra de passer du comportement défectueux au comportement normal. Ces développements pourraient être appliqués sur un véritable système manufacturier (http://www.univ-reims.fr/meserp/cellflex-4.0/cellflex-4.0,9503,27026.html) existant dans le laboratoire afin d'améliorer le travail proposé dans les futures recherches et de quantifier certains paramètres (réactivité, fausses alarmes, temps de cycle API…).

- Mehdi CHANKATE
  Titre des travaux : *Contribution à l'évaluation des approches de diagnostic par Model-Checking : Application aux Systèmes à Evénements Discrets.*
  Établissement et discipline : URCA, Génie Informatique, Automatique et Traitement du signal.
  Financement : Financement par son employeur (EMSI de Marrakech)
  Encadrement : V. Carré-Ménétrier (Directrice de Thèse) 33%, A. Philippot (Co-encadrant) 33%, P. Marangé (Co-encadrante) 33%.
  Première inscription à l'ED : 1er décembre 2016
  Date de soutenance : prévue en décembre 2019
  Production scientifique : 1 communication francophone [CN22].
  Remarque : Cette thèse fait également suite aux contacts obtenus après ma mobilité sortante en octobre 2015. La volonté de l'EMSI de Marrakech étant de créer un laboratoire d'automatique certifié.

Résumé :

Les travaux de recherche, relatés dans cette thèse, ont pour objectif de définir de nouvelles méthodes de vérification de la diagnosticabilité sur le modèle du système et sans définir un modèle diagnostiqueur. Un diagnostiqueur est basé sur la vérification de relations de conformité entre le fonctionnement désiré du système et la connaissance disponible sur ses fonctionnements normaux et anormaux, la plupart des méthodes de vérification trouvées dans la littérature, nécessitent la présence du *Diagnoser* (problème de construction de diagnostiqueur et l'algorithme de vérification de la diagnosticabilité). La méthodologie proposée permet de vérifier la non diagnosticabilité sans construction de diagnostiqueur ni de modèles intermédiaires. L'approche consiste à éviter la construction de modèles intermédiaires et par ailleurs les compositions d'automates en utilisant des instances de classe

du modèle G. La première instanciation G0 permet notamment de rechercher un cycle indéterminé et de comparer son observation à un autre cycle indéterminé issu de la seconde instanciation G1. Ainsi, sans composition parallèle, le *model-checker* permet de vérifier par équivalence en termes de parcours de graphe les conditions de diagnosticabilité.

3.3.3. Encadrements de Masters Recherche & Ingénieurs

- Hui Juan DIMANCHE (4 mois, d'avril à juillet 2013)
  Titre des travaux : *Contraintes de sécurité et diagnostic des SED. [CSA11]*
  Établissement et discipline : URCA, Master EEAII - Recherche.
  Encadrement : A. Philippot 100%
  Date de soutenance : 04 septembre 2013.

- Hanane SERRADJ (3 mois, de mai à juillet 2013)
  Titre des travaux : *Mise en place d'une approche de diagnostic sur systèmes manufacturiers à travers une méthode de classification des défaillances. [CSA10]*
  Établissement et discipline : Université des Sciences et de la Technologie d'ORAN « Mohamed Boudiaf », Algérie.
  Encadrement : A. Philippot 100%

- Emilien TIECHE (10 mois sur 2012-2013)
  Titre des travaux : *Commande et Supervision d'un atelier flexible. [CSA7, CSA8, CSA9]*
  Établissement et discipline : URCA, Ingénieur d'étude.
  Encadrement : B. Riera, F. Gellot, A. Philippot, D. Annebicque

3.3.4. Synthèse des encadrements doctoraux

Une synthèse chronologique des encadrements doctoraux depuis ma prise de fonctions d'EC est présentée en Figure 1-6. Si l'on s'y attarde un peu plus attentivement, il apparaît tout d'abord clairement une continuité dans les encadrements. Deuxièmement, 4 thèses sur 9 sont sur financement CIFRE, avec B. Riera comme directeur. Ce chiffre traduit notamment le fait que la recherche de l'équipe CDSED du CReSTIC est en lien avec les problématiques industrielles. La mobilité sortante demandée en 2015 m'a ainsi permis de nouer d'intéressants contacts avec le Maroc pour l'encadrement de thèses sous différentes sortes (cotutelle, financement employeur).

En considérant la verticalité de la Figure 1-6, et plus précisément l'année 2017 : 6 encadrements sont en parallèles, soit plus de 200% de taux d'encadrement. Ce chiffre

important fait suite à plusieurs facteurs dont notamment les 2 thèses CIFRE succédant aux 2 premières que j'avais encadrées.

Enfin, la durée moyenne des 6 doctorats soutenus encadrés sur l'ensemble de la thèse est de 37,3 mois. Il faut noter que la thèse au Maroc de Monsieur Y. Qamsane a duré 43 mois, l'ED de l'UCA Marrakech ayant pour habitude de soutenir des thèses en 4 ans. Concernant la thèse de Monsieur N. Malki, son directeur a cessé son encadrement avant la fin de la thèse. L'école doctorale a alors dû retrouver un directeur de thèse et j'ai effectué le co-encadrement sur les 6 derniers mois avec le Professeur Manamanni. Malgré une thèse en 57 mois, Monsieur Malki a trouvé un emploi suite au doctorat mais n'a pu achever la revue internationale attendue.

Les indicateurs de devenir des docteurs sont également très positifs puisque les 2 contrats CIFRE se sont traduits par un recrutement des docteurs au sein de l'entreprise (RATP et SNCF). Monsieur Malki, malgré une thèse très longue a lui aussi trouvé un emploi dans le domaine des services de technologie et d'ingénierie chez Sogeti. Monsieur Y. Qamsane souhaitait avoir une expérience à l'étranger : il a reçu 3 réponses positives de Post Doctorat et a choisi de rejoindre l'équipe du Professeure D. Tilbury de l'Université du Michigan qui travaille sur un sujet très proche de son doctorat. Enfin, les 2 prochaines thèses devant se soutenir fin 2018 (sous 36 mois) permettent de confirmer ce travail d'encadrements puisque Monsieur Niang a une promesse d'embauche pour janvier 2019 à la SNCF (suite à son contrat CIFRE) et Monsieur Pichard signe un CDI début janvier 2019 en tant qu'ingénieur en automatisme et informatique industrielle dans la société SASP Services.

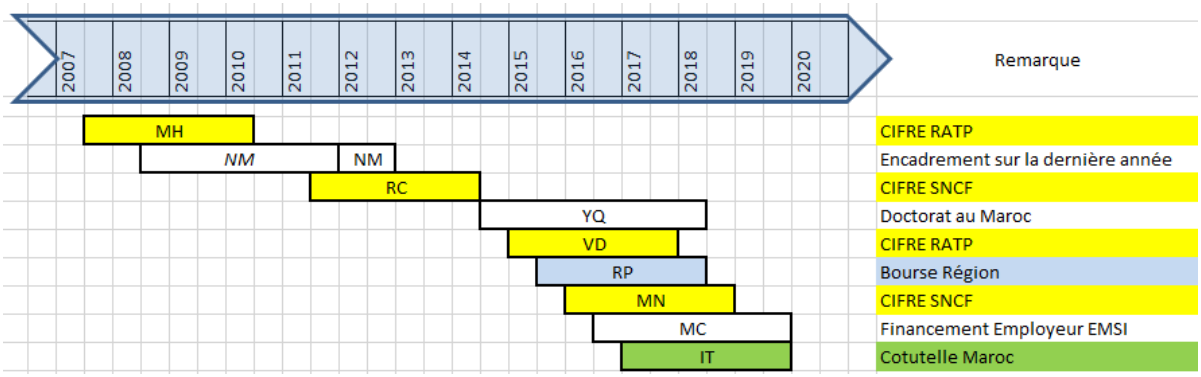

Figure 1-6 : Chronologie de l'encadrement doctoral

4. Responsabilités, Coopérations & Valorisation

Dans cette section, j'énumère brièvement les différentes missions que j'ai pu avoir sur le volet responsabilités. J'y intègre le côté valorisation (scientifique et pédagogique) mais aussi les coopérations et participations réalisées.

4.1. Vulgarisation Scientifique et Pédagogique

Que ce soit en Enseignement ou en Recherche, j'apporte une attention particulière aux activités de diffusion et de vulgarisation de la culture scientifique et technique de nos disciplines. A ce titre, j'interviens depuis 2007 lors des de différentes journées de diffusion (Fête de la Science, Classe en Fac, Collège des Sciences, Portes ouvertes, etc.). Lors de ces journées, nous accueillons, avec mes collègues, le grand public, des collégiens, des lycéens et même des primaires sur l'atelier flexible CellFlex. Nous adaptons notre discours et nos applications afin qu'ils puissent voir et également manipuler des systèmes automatisés.

La suite de logiciels de *Serious Games* de la société RealGames nous a permis de créer différents scénarios de manipulation. Nous avons par ailleurs fait une animation scientifique à la médiathèque Jean Falala de Reims en novembre 2008 (http://www.accustica.org/html/archives Agenda/agenda2008.html).

4.2. Responsabilités et Fonctions locales

Depuis 2012, je suis membre élu du Conseil de Laboratoire CReSTIC. Je participe ainsi à la vie du laboratoire et à ses décisions. Je suis par ailleurs co-responsable du Thème Transversal industrie 4.0 visant à fédérer les différentes équipes du CReSTIC pouvant travailler sur ce thème. Le plateau de formation et de transfert technologique CellFlex4.0 (http://www.univ-reims.fr/meserp/accueil/plateau-de-formation-et-de-transfert-technologique,9485,27019.html), dont je suis co-responsable, est une plateforme de formation et de recherche en lien direct avec les concepts d'usine du futur et d'industrie 4.0. Elle regroupe un ensemble d'outils pour la simulation de systèmes manufacturiers (ITS PLC, FACTORY I/O) et de maison virtuelle (HOME I/O), un atelier flexible (CellFlex) ainsi qu'une Plate-Forme Multi-Energies Renouvelables (PFMER). C'est un outil d'échanges pédagogiques, scientifiques et de vulgarisation. CellFlex4.0 a ainsi permis d'accueillir des délégations étrangères (République du Congo, Université d'Utah, délégation Québécoise) mais aussi des professeurs invités

comme le Dr Martin Fabian de l'Université de Chalmers en 2016. Des journées industrielles ont été organisées également avec 25 chefs d'entreprises membres du Comité des Jeunes Dirigeants (CJD) de la Marne.

Mes activités d'enseignement, de recherche et les relations avec le monde industriel m'ont permis d'être identifié au niveau de l'industrie 4.0 dans l'URCA. Le Président de l'Université m'a ainsi nommé **Chargé de Mission « Industrie du Futur »** en janvier 2018. Mon rôle est tout d'abord de coordonner l'action de l'université et du regroupement des établissements d'enseignement supérieur et de recherche dans le domaine de l'industrie du futur. Je suis en lien direct avec la chargée de mission « Suivi et développement des filières ingénieurs au sein de l'URCA », le pôle SNI, l'école doctorale Sciences du Numérique et de l'Ingénieur (SNI), et le Vice-Président de la Commission Recherche. Au titre de cette nomination, j'ai co-organisé deux journées techniques. La première a eu lieu en date du 10 avril 2018 avec un club d'industriels du Grand Reims (*Smart Factory Club*). Cette manifestation a réuni une soixantaine de personnes autour d'un programme mêlant cyber-sécurité et découverte du CReSTIC au travers de ces plateformes (https://smartfactoryclub.fr/?p=189). La seconde, avec 90 inscrits, a été co-organisé avec B&R Automation, ABB, FESTO et MappleSoft le 12 juin 2018 autour de la « Robotique et IoT Industriel ».

Au sein de l'établissement, une réflexion est menée pour la constitution d'un Institut « Services et Industries du Futur » de Reims, permettant de se positionner au sein d'un réseau Grand Est sur cette thématique. Cette réflexion s'appuie notamment sur la réalisation d'une plateforme « Industrie du Futur », à vision industrielle, en partenariat avec le Pôle Formation UIMM de Reims, et à vision scientifique, en partenariat avec l'UTT.

http://www.univ-reims.fr/universite/organisation/mission-industrie-du-futur,7741,18258.html?args=Yxp_PjQ3o54LWAz%252A0lEOsZT4oxnMK0PgGG2cE3hpq9FyqAHzXyYaSpCTMMlw6aow12wRMHllRxGKHMOGyYE0yA

4.3. Responsabilités et Fonctions dans les instances nationales et internationales

Au niveau national et international, je participe à différentes instances et sociétés savantes comme :

- Membre élu du Conseil National des Universités de la Section 61 (CNU61) au titre de suppléant depuis 2015 puis titulaire depuis septembre 2018.
- Membre élu du CA du Club EEA depuis 2013 (https://www.clubeea.org/).
- Secrétaire de la section « Automatique » du Club EEA depuis 2018.
- Vice-Président de la Commission Enseignement du Club EEA depuis 2018.
- Membre des comités de sélection (COS, CNU61) de l'Ecole Centrale de Lille (2011), de l'Université de Toulouse (2013 & 2016), de l'Université de Reims Champagne-Ardenne (2013).
- Représentant au comité technique "TC-3.1. Computers for Control" de l'IFAC "International Federation of Automatic Control" (depuis 2011).
- Reviewer pour plusieurs revues IFAC et IEEE (Automatica, CEP, DEDS, IJPR, IJSS, ISA Transaction, TAC, TII, …) et conférences nationales et internationales (Systol, MISC, 2SCS, MED, INCOM, IFAC WC, ICRA, CASE, DCDS...).

L'implication que j'ai au sein du Club EEA se traduit également par la co-organisation d'un concours national « Mon projet en 5 minutes » visant à promouvoir les projets étudiants dans la thématique du Club. Ce concours a été porté par la section automatique lors de la première édition en 2016, puis s'est élargi aux autres sections (Electronique, Electrotechnique, Signal et Image), et à la commission enseignement du Club EEA.

4.4. Participation à des Communautés Scientifiques

Au niveau national, depuis le début de mon doctorat, je participe régulièrement au groupe de travail SED (Systèmes à Evénements Discrets) anciennement INCOS (Ingénierie de la Commande et de la Supervision des SED) du GdR-MACS (Groupe de Recherche sur la Modélisation Analyse et Conduite de Systèmes dynamiques). En 2010, j'ai présenté un projet de travail commun autour d'un benchmark simulé pour l'étude et la comparaison d'approches de diagnostic des SED. Ce travail a abouti à la proposition d'une session spéciale lors du Workshop DCDS'11 (*International workshop Dependable Control of Discrete Systems*). J'ai également participé à quelques réunions d'autres GT comme S3, Bermudes, IMS2 lors des journées STP (Sciences et Techniques de la Production) ou Automatique du GdR MACS.

Au niveau international, j'ai participé au premier Workshop on Product Intelligence en septembre 2012 qui s'est tenu à l'University of Cambridge (UK). Même si de colloque est

quelque peu éloigné de la thématique de la commande par le produit, cette rencontre a permis de montrer les activités du laboratoire. Sur un autre plan, j'ai collaboré lors de plusieurs réunions au GDR International HAMASYTI (*International Research Network on HumAn-MAchine SYstems in Transportation and Industry*) en 2014 et 2015. Ce GDRI vise à créer un réseau de recherche autour de la conception, de l'organisation et de la validation de systèmes homme-machine et de leurs applications dans le domaine des transports et de l'industrie (http://www.cnrs.fr/ins2i/spip.php?article1805).

4.5. Participation à des Jurys de Thèses extérieurs

- Examinateur de la thèse de M. Matthieu PERIN, Contribution à la modélisation réaliste et multi-échelles des systèmes bouclés temporisés, LURPA - ENS Cachan (soutenue le 22 juin 2012).
- Examinateur de la thèse de M. Baisi LIU, An Efficient Approach for Diagnosability and Diagnosis of DES based on Labeled Petri Nets: Application to Railway Safety, Ecole Centrale de Lille (soutenue le 17 avril 2014).

4.6. Membre de Comités de Programmes ou d'Organisation

- S-mart 2019 : 16ème Colloque National S-mart (Anciennement AIP Primeca), Les Karellis, avril 2019.
- WCCS14 : 2nd World Conference on Complex Systems, Agadir, Maroc, 2014.
- 2SCS12 : International Symposium on Security and Safety of Complex Systems, Agadir, Maroc, 2012.
- ICCS12 : International Conference on Complex Systems, Agadir, Maroc, 2012.
- MISC11 : Conférence Méditerranéenne sur l'ingénierie sûre des systèmes complexes, Agadir, Maroc, 2011.
- ICONS16 : 4th IFAC International Conf. on Intelligent Control and Automation Sciences, Reims, France, 2016.
- EAM09 : 28th European Annual Conf. on Human Decision-Making and Manual Control, Reims, France, 2009.
- DCDS07 : 1st IFAC workshop Dependable Control of Discrete Systems, Cachan-Paris, France, 2007.

4.7. Organisation de Journées Scientifiques et Sessions Spéciales

- BWMMS15 : Special Session "Advanced non-intrusive tools to diagnose and

support human activities" - 11th Berlin Human-Machine Systems Workshop, Berlin, Germany, 2015 (co-organisée avec Ph. Polet).

- Safeprocess15 : Special Session "From Control to Diagnosis of Discrete Event Systems" – 9th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes, Paris, France, 2015 (co-organisée avec P. Marangé).
- DCDS15 : Special Session "Diagnosis approaches: What news since over 20 years?" – 5th International workshop Dependable Control of Discrete Systems, Cancun, Mexico, 2015.
- Séminaire Scientifique lors d'une mobilité sortante pour l'Université de Cadi Ayyad (Maroc) en octobre 2015.
- 2SCS12 : Invited Session "From Control to Diagnosis of Discrete Event Systems" - International Symposium on Security and Safety of Complex Systems, Agadir, Morocco, 2012.
- DCDS11 : Special Session "Diagnosis of Discrete Event Systems: Application on a Benchmark" - 3rd International workshop Dependable Control of Discrete Systems, Saarbrücken, Germany, 2011.
- DCDS09 : Special Session "DES Diagnosis" - 2nd IFAC workshop Dependable Control of Discrete Systems, Bari, Italy, 2009.

4.8. Organisation et Encadrements d'échanges internationaux

En 2009 et 2010, le département EEA de l'URCA à participer à l'*European Team Seminar*. J'ai été l'organisateur et encadrant côté Rémois de ce programme. Il entre dans le cadre d'un projet Européen entre l'Université Technique de Kaiserslautern (Germany) et l'Université de Reims Champagne-Ardenne. Chaque année universitaire, une problématique industrielle était posée, courant novembre, à une dizaine d'étudiants allemands et autant Rémois. Les étudiants allemands venaient ensuite 5 jours à Reims en février afin présenter leurs avancements. Le projet se poursuivait quelques jours chez nos voisins outre-Rhin afin de finaliser le projet et présenter une solution devant l'industriel. Les étudiants ont pu durant ces différents séjours visiter différents sites industriels (Siemens Turbomachinery Equipment GmbH, Mann+Hummel Group, Nicolas Feuillate), mais aussi les plateaux techniques de chaque ville : CellFlex pour Reims et la *SmartFactory* de Kaiserslautern (vitrine technologique établissant le lien entre la recherche scientifique et l'industrie).

Ce type d'événement a été reconduit lors de l'Erasmus Intensive Program HUMAIN (*HUman - MAchine INteraction*). Ce programme regroupe quatre universités (Salerne (It), Valence (Esp), Kaiserslautern (All) et Reims (Fr)). C'est un projet éducatif qui a pour tâche principale de proposer un défi industriel autour de l'interaction homme-machine. Il s'agit aussi

de relations internationales et interdisciplinaires, offrant aux étudiants qui participent au programme une opportunité unique à ajouter à leur curriculum vitae. Après avoir fait participer nos étudiants sur les éditions de Salernes et Valences, Bernard Riera, François Gellot et moi-même, avons organisé sur Reims l'édition 2013. Cet échange international consiste à créer 6 groupes de 4 étudiants de chaque pays. Ils se réunissent deux semaines afin d'apprendre, de travailler et de profiter ensemble. Le challenge 2013 a été de proposer une solution à la robotisation de l'étape d'entreillage de bouteilles dans les caves du Centre Vinicole - Champagne Nicolas Feuillatte**.**

Dans le cadre d'une mobilité entrante, le CReSTIC a eu la chance d'accueillir le Professeur Martin Fabian de l'Université de Chalmers en février 2016 (http://www.univ-reims.fr/actualites-presse/actualites-presse,9499,27020.html). Lors de sa visite, l'équipe CDSED a pu échanger avec M. Fabian sur nos travaux autour de la théorie de supervision. Il a également présenté ces travaux de recherche autour des SED durant un séminaire. Cet échange doit aboutir à la soumission d'un article commun dans une revue internationale.

4.9. Participation à des Contrats de Recherche (autres que CIFRE)

Je détaille dans cette section les différents projets auxquels j'ai pu contribuer. Auparavant, on pourra retrouver en Tableau 1-3 une synthèse chiffrée de ceux-ci.

	Type	**Années**	**Nbr. intervenants**	**Part CReSTIC**
MOSYP	CPER	2007-2013	>20	300 k€
EDUCASCOL	CPER	2007-2013	7	81,5 k€
ADEXEC	GIS	2011-2012	9	15 k€
AI 1	Action Incitative	2012	2	6 k€
DOODIE	Action Incitative	2014	2	6 k€
SUCRé	Inter-Région	2015-2017	>20	136 k€
Virtual Commissioning	GIS S-mart	2018	5	10 k€/12 k€
HUMANISM	ANR	2017-2020	14	15 k€/310 k€

Tableau 1-3 : Synthèse des projets

4.9.1. Mesures des performances et Optimisation des SYstèmes de Production (MOSYP)

MOSYP est un Contrat de Projets Etat-Région (CPER 2007-2013) entre deux laboratoires, le CReSTIC de l'URCA et le LOSI de l'UTT, financé en partie par la Région Champagne-Ardenne, la ville de Reims, l'Etat et quelques industriels. Il concerne les activités liées au contrôle/commande et à l'ordonnancement de l'atelier de production. Le projet MOSYP s'appuie sur le plateau technique MESERP composé entre autres des trois cellules flexibles sur Troyes et Reims et de la plate-forme d'optimisation TOA-ECL à Troyes. C'est un projet devant fournir des outils conjuguant différents modèles en vue d'optimiser des critères pouvant être antagonistes. Il explore, également, le contexte dans lequel les indicateurs de performance se développent ainsi que les conséquences, de leur mise en place sur l'optimisation d'un système de production. Les méthodologies, les algorithmes ainsi que les outils d'analyse et d'évaluation à développer intègrent les points de vue sociaux, économiques, technologiques et environnementaux.

Dans le cadre de ce projet, j'ai été responsable de la Tâche « Diagnostic et Surveillance ». Par ailleurs, j'ai grandement participé à la mise en place de l'extension robotique de la cellule flexible. Dans le cadre du projet MOSYP, une restitution des travaux a été organisée le 18 Novembre 2013 devant une trentaine d'invités. J'ai publié dans les différentes tâches du projet. L'ensemble des publications peuvent être retrouvé ici : http://www.univ-reims.fr/meserp/projets/publications/publications,14799,25558.html.

4.9.2. Education cognitive et scolarisation (EDUCASCOL)

EDUCASCOL est un projet régional du CPER 2007-2013 visant à lutter contre l'échec scolaire en « outillant » les élèves dès les premiers cycles. Il s'inscrit dans l'axe « Dynamique des territoires » du pôle de recherche SHS, plus particulièrement du point de vue de la dynamique des territoires éducatifs, a pour objectif de répondre aux besoins régionaux qui traduisent un problème sociétal tant national qu'européen, à savoir l'échec scolaire. Il s'agit de développer des recherches qui permettront de favoriser à terme les pré-requis de la réussite scolaire, dès les premières années de scolarisation, en ciblant les mécanismes cognitifs (projet 1) et langagiers (projet 2) à l'origine du développement conceptuel habituel qui autorisent une

entrée harmonieuse dans les apprentissages. Il a été le fruit d'une association avec une équipe de recherche en psychologie cognitive de l'URCA, l'équipe labellisée CLEA, qui développe des études sur les facteurs de prévention et de remédiation des troubles cognitifs qui sont responsables de l'échec scolaire.

Côté participation du CReSTIC, une première opération consistait à réaliser un dispositif expérimental permettant aux enfants de réaliser une tâche de pilotage d'un système automatisé. L'objectif était de déterminer les ressources cognitives liées à l'apprentissage mises en œuvre par les enfants. Un premier protocole expérimental, proposé par l'équipe du CReSTIC, a rapidement été mis en place au début du projet. Les premières expérimentations par les psychologues ont montré que ce protocole était lourd à utiliser pour eux. Mais, surtout, ce premier protocole a mis en évidence que pour les enfants issues de classes de CE1 et de CP l'exercice était relativement complexe et qu'aucun enfant de CP n'avait répondu juste à l'expérimentation. Cette première phase expérimentale a par conséquent permis de définir un âge minimal requis pour cet exercice, soit 8 ans. Le projet a financé ensuite un contrat post-doctoral au CReSTIC, David Annebicque. Il a travaillé essentiellement sur la commande sûre de fonctionnement à base de contraintes, mais aussi sur un logiciel permettant de simuler et d'interpréter un GRAFCET écrit grâce au logiciel SFCEdit proposé par Monsieur Di-Meglio avec qui le CReSTIC a noué un partenariat. L'intérêt de ce logiciel, l'InterpréteurG7 est surtout de pouvoir faciliter le travail de recherche du CReSTIC en n'étant plus dépendant d'une solution propriétaire. Ce logiciel a aussi permis de simplifier de manière significative le protocole expérimental original. Le travail a fait l'objet d'une publication dans un CETSIS [CN10].

4.9.3. Approche de Détection et d'EXplication d'Erreur de Commande par filtrage robuste

Ce projet GIS (2011-2012) se proposait d'étudier les systèmes critiques, pilotés par des opérateurs humains, qui nécessitent une aide à la conduite. Ce projet avait pour objectif de proposer une approche assurant la sécurité des personnes et de l'environnement, ainsi que le « bon » pilotage du système en mode normal et en mode dégradé. C'est une collaboration avec le LAMIH de l'Université de Valenciennes et le CRAN de l'Université de Lorraine. Pour répondre à cette problématique, plusieurs verrous scientifiques ont été identifiés :

- La mise en place d'une approche formelle pour développer les filtres de commande tout en respectant un niveau de SIL (*Safety Integrity Level*) donné pour ces fonctions de sécurité. Par exemple à partir du niveau SIL3, la sécurité doit être prouvée formellement. Quels sont les modèles et la méthode adaptés au contexte normalisé encadrant le développement de fonctions de sécurité ?
- Le filtrage robuste va dépendre des informations provenant de la commande, sur lesquelles le filtre peut agir mais aussi des informations provenant de la partie opérative. Le résultat du filtrage est fortement dépendant de la fiabilité de la PO. Comment prendre en compte les éventuelles défaillances de la PO et les interactions avec les processus de diagnostic des installations ?
- La génération des explications à destination d'un opérateur humain doit être compréhensible et suffisamment synthétique vis à vis de son niveau d'expertise. Quels sont les mécanismes d'abstraction ou de synthèse permettant de générer une explication compréhensible ?

L'interaction entre le filtre de commande et le diagnostic doit permettre de considérer l'état réel du système, c'est à dire les possibles défaillances des capteurs ou des actionneurs. En effet, le fait qu'un défaut survienne sur un capteur ou sur un actionneur, implique qu'il n'est plus possible de lui faire confiance. Par conséquence, il faut en tenir compte du fait qu'il y ait un défaut ou non, dans la définition du filtre. Chaque partition de défauts aura un bit pour informer le filtre sur l'état de décision du diagnostic. Cette information permettra soit d'utiliser la variable si l'information de défaillance est à 0, soit d'utiliser son équivalence si cette information de défaillance est à 1. Lors d'une défaillance, il est possible d'utiliser une information équivalente seulement pour les capteurs, en effet lorsqu'un actionneur est défaillant soit il faut effectuer une tâche de maintenance, soit il faut avoir une redondance fonctionnelle de celui-ci. Pour définir l'information équivalente d'un capteur, nous nous sommes basés sur l'information temporelle qui sépare l'évolution d'une commande de celle d'un capteur et elle a été ajoutée dans les contraintes.

Concernant la génération d'explications, les premiers travaux réalisés ont consisté à cerner de manière exacte le rôle des opérateurs humains. Hormis l'opérateur de conception – celui qui crée la commande qui peut être validée hors ligne – deux « types » d'opérateurs peuvent être distinguées : i) un opérateur de production, en lien direct avec la partie opérative, ii) un

opérateur de maintenance en charge de corriger la commande si celle-ci s'avère défaillante.

Les tâches de ces deux types d'opérateurs sont différentes et peuvent conduire à des utilisations différentes du filtre tel qu'il est défini actuellement. Pour ce projet, seul l'opérateur de maintenance a été considéré. Son rôle essentiel consiste ici à assurer le diagnostic de la commande pour ensuite la corriger. Dans ce cadre, le filtre de sécurité assure le blocage de la partie opérative, i.e. le positionnement des actionneurs, de manière à empêcher toutes requêtes erronées issues de la partie commande. Les éléments accessibles au système d'aide pour assurer l'explication à cet opérateur de maintenance sont la règle de sécurité déclenchée, et les entrées et sorties de l'automate qui assure l'exécution de la commande. Les programmes automates ainsi que ses éléments internes tels que temporisation et mémoire sont considérés comme non accessibles, leur analyse étant trop problématique. L'aide au diagnostic de la commande qui peut être faite n'est alors que partielle. Ce projet a fait l'objet d'un poster lors du 4ème Workshop du Groupement d'Intérêt Scientifique « Surveillance, Sûreté, Sécurité des Grands Systèmes » (GIS-3SGS'11) à Valenciennes, mais aussi de 2 conférences internationales [CI29, CI30].

4.9.4. Actions Incitatives

En 2012, j'ai participé à projet « Action Incitative » avec le CRAN (porteur P. Marangé, CRAN) sur la suite de l'interaction Filtre/Diagnostic du projet ADEXEC. En effet, afin d'assurer la robustesse du filtre en mode dégradé, face aux erreurs de commande, une approche de vérification par *model-checker* a été mise en place. Cette vérification formelle a nécessité la modélisation de la partie opérative, des diagnostiqueurs, du filtre, de la commande la plus permissive et de l'environnement de calcul. Ce travail a permis de constituer un sujet de thèse débuté par Mehdi Chankate fin 2016.

J'ai également participé à un second projet « Action Incitative » URCA en 2014 (porteur R. Saddem). Ce projet DOODIE (DiagnOsis Of DIscrete Events) consistait à montrer la principale limite de l'approche diagnostiqueur proposée dans mon doctorat. En effet, l'approche classique consiste à décomposer le système en composants. Ensuite, un modèle automate est associé à chaque composant. Le diagnostiqueur global du système dans une approche centralisée est alors obtenu par le produit synchrone des automates de ses

composants. C'est ce produit synchrone qui engendre souvent une explosion combinatoire quand le système est complexe. Pour faire face à cette problématique, des travaux sont en cours dans deux directions : d'une part la conception de diagnostiqueurs décentralisés (Debouk *et al.*, 2000) ou distribués (Su, 2004), d'autre part la construction d'observateurs basés sur des règles expertes comme les Signature Temporelle Causale (STC) (Saddem *et al.*, 2012). Une STC est un sous-ensemble d'événements observables partiellement ordonnés qui caractérisent un comportement défaillant d'un système. Ces événements sont contraints par un ensemble de contraintes temporelles portant sur leurs occurrences. En pratique, une STC est la description d'un motif temporel définissant un ordre partiel, sur des événements décrits par leur type et leur date d'occurrence. Les relations entre les événements peuvent être logiques (conjonction) ou temporelles (séquence, absence …).

4.9.5. Sûreté de fonctionnement et résilience pour la gestion et le contrôle coopératif des systèmes sociotechniques (SUCRé)

SUCRé (Sûreté de fonctionnement et résilience pour la gestion et le contrôle coopératif des systèmes sociotechniques) est un projet inter-régional entre les régions Nord-Pas-de-Calais et Champagne-Ardenne - Coopération Homme(s)-Robot(s) en milieu hostile (2015-2017). L'objectif de ce projet est le développement d'outils, méthodologiques et technologiques, pour la gestion et le contrôle de situations de crises en termes de résilience et ce, avec une approche originale et pluridisciplinaire. Les systèmes considérés sont des systèmes sociotechniques, comme les organisations militaires ou de sécurité civile prenant en charge la gestion de crises et reposant sur des coopérations entre les structures de commande, les opérateurs humains de terrain et l'ensemble des ressources mises à leur disposition. Parmi ces ressources, en milieu hostile, des robots mobiles peuvent être utilisés, permettant que les opérateurs humains ne soient pas exposés au danger. Ces robots peuvent être dotés de capacités de communication et de coopération, augmentant ainsi l'efficacité du système socio technique considéré. Le projet SUCRé se focalise sur l'étude et la mise au point d'outils d'aide à la coopération Homme(s)-Robot(s) au sein de systèmes sociotechniques, évoluant dans des environnements complexes et dynamiques. De ce fait, la coopération entre les opérateurs humains ainsi que celle entre les humains et les robots, doit être étudiée et modélisée dans ce type d'environnements et ce, afin de parvenir à la mise au point de

systèmes d'aide à la coopération, la décision et la planification. Par ailleurs, la coopération robot-robot, qui se base sur une auto organisation des agents robotisés pour répondre à une mission confiée par la cellule de crise, est supervisée par un opérateur humain déporté. Ce sont des problématiques particulières en plein développement dans le domaine de l'interaction Homme-Robots qui reposent sur la prise en compte d'un certain nombre de verrous scientifiques relevant de l'automatique, de l'informatique et de la psychologie cognitive. Dans le cadre de ce projet, j'ai été co-responsable du Livrable 1 Spécification des Besoins.

4.9.6. Projet GIS S-mart

Le projet GIS S-mart, *virtual commissioning* pour une pédagogie active sur la commande des systèmes cyber-physiques, est un projet sur 11 mois en 2018. Ce projet vise à développer une pédagogie innovante avec des outils numériques (propres à l'automatique) en interconnexion avec un jumeau numérique, autour d'une plateforme permettant d'avoir une vision systémique multi points de vue du système de production. Il doit faire ainsi le lien entre différents modules pédagogiques. Ce projet est en partenariat avec des collègues de 3 établissements (URCA, UVHC, UL) dans les domaines de la formation au contrôle/commande, la synthèse de la commande sûre de fonctionnement, la coopération Homme-Machine et la simulation de Parties Opératives.

4.9.7. ANR HUMANISM

Le projet ANR HUMANISM « Coopération Hommes-Machines pour des systèmes de production flexibles » a débuté en décembre 2017 (42 mois), le responsable scientifique est Damien Trentesaux (Université de Valenciennes).

L'un des objectifs de l'Usine du Future est de fabriquer rapidement des produits sur mesure, de haute qualité, avec un niveau de consommation énergétique bas en amenant de la souplesse au sein du système manufacturier global. Cet objectif implique de construire des systèmes manufacturiers plus flexibles et plus résilients. L'intégration de nouvelles technologies issues des domaines de la productique et du numérique constitue une piste intéressante. Cependant, cette évolution doit s'accompagner d'une démarche qui maintient l'Homme au centre du système global, tant sur les aspects décisionnels que sur le contrôle des

systèmes « intelligents » implémentés. C'est dans ce contexte que HUMANISM propose de mettre en place une approche de conception centrée sur l'Homme. L'idée majeure est d'équilibrer les implications de l'Homme et de la technologie, en profitant des avancées de l'automatisation, mais également des capacités d'adaptation de l'Homme, ainsi que de ses capacités à tirer profit de son expérience. Les systèmes manufacturiers dits « intelligents » (SMI) déploient des Systèmes Auto-Organisés (SAO) efficients, mais qui sont à même de générer des comportements émergeants pouvant conduire à des situations inattendues et dangereuses ; HUMANISM a pour objectif de développer une méthodologie de conception de systèmes d'aide coopératifs, de façon à soutenir la conscience de la situation de l'Homme, ainsi que ses prises de décisions pour le contrôle de tels SMI.

L'approche de conception centrée sur l'Homme proposée dans HUMANISM repose, d'une part, sur l'intégration d'une méthode d'Analyse de l'Activité Cognitive (AAC). Il s'agit de l'une des principales méthodes qui prend en considération les facteurs humains pour la conception de systèmes sociotechniques. Cette méthode pourra assister la conception de système d'aide pour les SMI en fonction des besoins actuels et futurs de l'Homme, besoins qui seront redéfinis de par l'implantation de nouvelles technologies telles que les SAO. D'autre part, l'approche de conception centrée sur l'Homme repose sur les principes de la Coopération Homme-Machine (CHM) qui propose des modèles précis permettant d'identifier et d'implémenter des organisations et des partages de tâches optimaux entre l'Homme et les nouvelles technologies. C'est dans le but d'assurer la généricité de nos développements théoriques que HUMANISM testera ses solutions sur trois systèmes manufacturiers. Ils mettront en œuvre des technologies et des modèles de comportement différents au regard de leur capacité d'auto-adaptation et des niveaux d'automatisation qu'il serait possible d'implémenter. Les trois SAO sont un cobot, une patrouille de robots mobiles et une patrouille de produits intelligents. Ils se différencient également du point de vue de leur interaction avec l'Homme (proche vs. distant) et dans leur comportement plus ou moins prédictif, ou aussi dans leur capacité de communication. Chaque SAO est dès à présent disponible dans leur version réelle ou simulée dans les laboratoires impliqués dans HUMANISM. Je participe actuellement au livrable 1 sur l'état de l'art en diagnostic et suis responsable du livrable 2 : Définition des Uses Case.

5. Publications

Ce paragraphe vient conclure ce chapitre par l'ensemble des publications (scientifiques comme pédagogiques). Celles-ci sont également disponibles sur ma page web du laboratoire : https://crestic.univ-reims.fr/fr/alexandre.philippot. Dans cette liste, les noms en italiques représentent les doctorants et les étudiants en master encadrés au cours de la carrière. Le bilan quantitatif est repris en Tableau 1-4. On constate que 54% des communications sont des conférences internationales (Figure 1-7). Ce taux élevé s'exprime par politique du laboratoire qui privilégiait davantage l'échange lors de rencontres que les revues. La politique a évoluée avec les critères d'évaluation nationale des laboratoires et de valorisation de la recherche (Figure 1-8). Il est à noter que les communications nationales ne sont pas délaissées et permettent notamment une valorisation des activités pédagogiques.

Type de publication	
Revues internationales (RI / RI JCR)	9 / 6
Revues nationales (RN)	5
Ouvrages et Chapitres d'ouvrages (CH)	2
Conférences internationales (CI)	57
Conférences nationales (CN)	22
Rapports et communications sans acte (CSA)	14

Tableau 1-4 : Bilan quantitatif de la production scientifique (période 2003-2018)

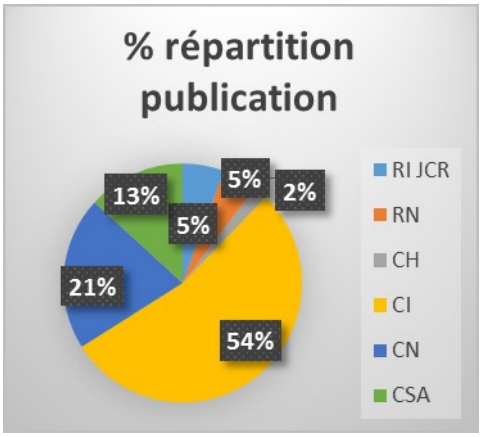

Figure 1-7 : Répartition des publications

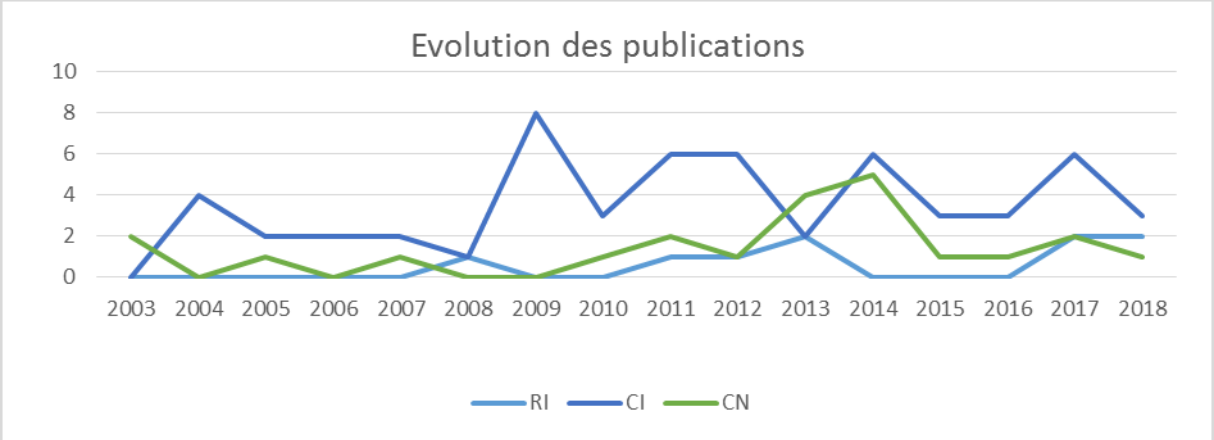

Figure 1-8 : Evolution des publications (RI vs CI vs CN)

Selon la source *Google Scholar*, ces travaux montrent une évolution stable des citations dans la communauté avec une moyenne de 24 citations par an (Figure 1-9). Exception faite pour l'année 2017 avec 53 citations. Une explication plausible à cette hausse est la publication de 2 revues internationales. Mes travaux les plus cités restent encore mon mémoire de doctorat et 2 publications sorties juste après sur le thème du diagnostic des SED. La source ResearchGate indique quant à elle, à la date du 22/09/2018, 2 849 lectures sur les dernières 36 semaines et 171 citations.

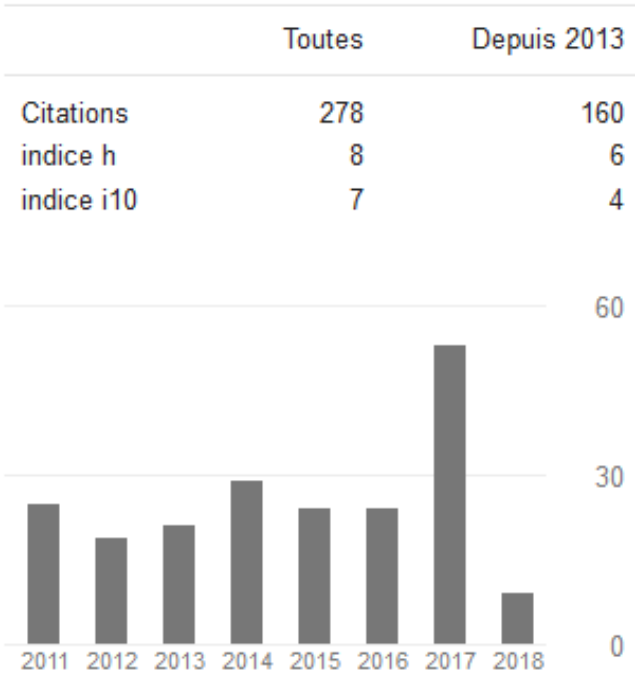

Figure 1-9 : Citations (source Google Scholar au 22/09/2018)

REVUES INTERNATIONALES AVEC COMITE DE LECTURE :

[RI9] ***R. Coupat*, A. Philippot, *M. Niang*, C. Courtois, D. Annebicque and B; Riera**. Methodology for Railway Automation Study and Automatic Generation of PLC Programs. IEEE Intelligent Transportation Systems Magazine (ITSM), Issue 99, Vol. PP. 2018. 10.1109/MITS.2018.2842018 (JCR - 2016 Impact Factor: 3.654), (Q1 en 2017 dans Scimago Journal & Country Rank)

[RI8] ***R. Pichard*, A. Philippot, R; Saddem and B. Riera**. Safety of Manufacturing Systems Controllers by Logical Constraints With Safety Filter. IEEE Transactions on Control Systems Technology (TCST), Issue 99, Vol. PP. 2018. 10.1109/TCST.2018.2827329 (JCR - 2016 Impact Factor: 3.882), (Q1 en 2017 dans Scimago Journal & Country Rank)

[RI7] ***Y. Qamsane*, A. Tajer and A. Philippot**. Towards an approach of synthesis, validation and implementation of distributed control for AMS by using events ordering relations. International Journal of Production Research (IJPR), Issue 21, Vol. 55. 2017. https://doi.org/10.1080/00207543.2017.1333648 (JCR - 2016 Impact Factor: 2.325), (Q1 en 2017 dans Scimago Journal & Country Rank)

[RI6] ***Y. Qamsane*, A. Tajer and A. Philippot**. A synthesis approach to distributed supervisory control design for manufacturing systems with GRAFCET implementation. International Journal of Production Research (IJPR), Issue 15, Vol. 55. 2017. http://dx.doi.org/10.1080/00207543.2016.1235804 (JCR - 2016 Impact Factor: 2.325), (Q1 en 2017 dans Scimago Journal & Country Rank)

[RI5] **A. Philippot, P. Marangé, F. Gellot and B. Riera**. Decentralized Diagnosis and Diagnosability by Model Checking. Horizon Research Publishing Corporation, Universal Journal of Control and Automation, 1(2):28-33, September 2013. (ICDS = 0.301),

[RI4] **A. Tajer, A. Philippot and V. Carré-Ménétrier**. Centralised controller for manufacturing systems through liveness extraction approach. Inderscience publishers, International Journal of Systems, Control and Communications, 5(3):189-213, 2013. (Indexed in Scopus), (Q2 en 2013 dans Scimago Journal & Country Rank)

[RI3] **A. Philippot, A. Tajer and V. Carré-Ménétrier**. From centralized to decentralized approach for optimal controller of Discrete Manufacturing Systems. ARPN Journal of Science and Technology, 2(10):936-949, November 2012.

[RI2] **A. Philippot, M. Sayed Mouchaweh, V. Carré-Ménétrier and B. Riera**. Generation of candidates' tree for the fault diagnosis of discrete event systems. Special Section: DCDS'09, Elsevier, Control Engineering Practice, 19(9):1002-1013, September 2011. (JCR - Impact Factor: 1.814), (Q1 en 2011 dans Scimago Journal & Country Rank)

[RI1] **M. Sayed Mouchaweh, A. Philippot and V. Carré-Ménétrier**. Decentralized

Diagnosis by Boolean Discrete Event System Model: Application on manufacturing systems. International Journal of Production Research (IJPR), Special Issue, 46(19):5469-5490, septembre 2008. 10.1080/00207540802367074. (JCR - Impact Factor: 1.477), (Q1 en 2008 dans Scimago Journal & Country Rank)

REVUES NATIONALES AVEC COMITE DE LECTURE :

[RN5] ***R. Coupat*, B. Riera, C. Courtois, M. Meslay, A. Philippot et D. Annebicque.** Méthodologie pour les études d'automatisation et la génération automatique de programmes API. Revue générale des chemins de fer, Hervé Chopin (HC Éditions), 254:28-40, novembre 2015.

[RN4] **P. Marangé, S. Debernard, F. Gellot, M. Pacaux, A. Philippot, T. Poulain, B. Riera et J.-F. Pétin.** Approche de détection et d'explication d'erreur de commande par filtrage robuste. Hermès/Lavoisier, Journal Européen des Systèmes Automatisés, 48(4):339-372, décembre 2014. (Indexée dans Scopus)

[RN3] **A. Philippot, M. Sayed Mouchaweh et V. Carré-Ménétrier.** Diagnostic Décentralisé des SED: Application aux Systèmes Manufacturiers. Hermès/Lavoisier, Journal Européen des Systèmes Automatisés JESA, 42(1):31-62, mars 2008. (Indexée dans Scopus)

[RN2] **A. Philippot, A. Tajer, F. Gellot et V. Carré-Ménétrier.** Méthodologie de modélisation dans le cadre de la synthèse formelle des SED. Revue électronique des sciences et technologies de l'automatique, e-STA, 2(2), mai 2005.

[RN1] **A. Philippot, A. Tajer, F. Gellot et V. Carré-Ménétrier.** Approche de synthèse en ligne basée sur une modélisation structurée de la partie opérative. Revue électronique des sciences et technologies de l'automatique, e-STA, 1(3), 2004.

OUVRAGES ET CHAPITRES D'OUVRAGES :

[CH2] **A. Philippot**. Diagnostic Décentralisé des Systèmes à Evénements Discrets. Éditions Universitaires Européennes, ISBN 9786131524653, 2010.

[CH1] **A. Philippot, M. Sayed Mouchaweh and V. Carré-Ménétrier**. Chapter 16: Component models based approach for failure diagnosis of Discrete Event Systems. Intelligent Industrial Systems: Modelling, Automation and Adaptive Behaviour, IGI, ISBN 9781615208494, 2010.

CONFERENCES INTERNATIONALES :

[CI57] ***R. Pichard*, A. Philippot and B. Riera**. Safe PLC Controller implementation IEC

61131-3 compliant based on a simple SAT solver: Application to manufacturing systems. 15th International Conference On Informatics in Control, Automation and Robotic (ICINCO'18), Porto, Portugal, July 2018.

[CI56] **A. Philippot, B. Riera, *V. Kunreddy* and S. Debernard**. Advanced Tools for the Control Engineer in Industry 4.0. The 1st IEEE International Conference on Industrial Cyber-Physical Systems (ICPS 2018), Saint-Petersburg, Russia, May 2018.

[CI55] ***I. Tahiri*, A. Philippot, V. Carré-Ménétrier and A. Tajer**. Timed synthesis control approach for tolerant-fault control of Discrete Event Systems (DES). The International Conference on Control, Automation and Diagnosis (ICCAD'18), Marrakech, Morocco, March 2018, *Best Student Paper Award*.

[CI54] ***M. Niang*, A. Philippot, F. Gellot, R. Coupat, B. Riera and S. Lefebvre**. Formal Verification for Validation of PSEEL's PLC Program. 14th International Conference On Informatics in Control, Automation and Robotic (ICINCO'17), Madrid, Spain, July 2017.

[CI53] ***R. Pichard*, A. Philippot and B. Riera**. Consistency Checking of Safety Constraints for Manufacturing Systems with Graph Analysis. 20th IFAC World Congress 2017 (IFAC WC), Toulouse, France, July 2017.

[CI52] ***V. Dimanche*, A. Goupil, A. Philippot, B. Riera, A. Urban and G. Gabriel**. Massive Railway Operating Data Visualization; a Tool for RATP Operating Expert. 20th IFAC World Congress 2017 (IFAC WC), Toulouse, France, July 2017.

[CI51] ***Y. Qamsane*, M. El Hamlaoui, A. Tajer and A. Philippot**. A Tool Support to Distributed Control Synthesis and GRAFCET Implementation for Discrete Event Manufacturing Systems. 20th IFAC World Congress 2017 (IFAC WC), Toulouse, France, July 2017.

[CI50] **A. Philippot, B. Riera, M. Koza, *R. Pichard*, R. Saddem, F. Gellot, D. Annebicque and F. Emprin**. HOME I/O and FACTORY I/O - 2 Pieces of innovative PO simulation software for automation education. The 27th European Association for Education in Electrical and Information Engineering Annual Conference (EAEEIE 2017), Grenoble, France, June 2017.

[CI49] ***Y. Qamsane*, M. El Hamlaoui, A. Tajer and A. Philippot**. A Model-Based Transformation Method to Design PLC-Based Control of Discrete Automated Manufacturing Systems. 4th International Conference on Automation, Control Engineering & Computer Science (ACECS'17), Tanger, Morocco, Proceedings of Engineering and Technology – PET Vol.19, pp.4-11, March 2017.

[CI48] ***Y. Qamsane*, A. Tajer, A. Philippot and A. Elbacha**. Synthesis and Implementation of Timed Distributed Supervisory Controller: Application to an Automated Manufacturing System. 3rd International Afro-European Conference for Industrial Advancement (AECIA 2016), Marrakech, Morocco, November 2016. *Best*

Paper Award of AECIA 2016.

[CI47] ***R. Coupat*, A. Philippot, D. Annebicque, F. Gellot and B. Riera**. Automation for improve of mental workload of the systems engineer. The 13th IFAC/IFIP/IFORS/IEA Symposium on Analysis, Design, and Evaluation of Human-Machine Systems (IFAC HMS 2016), Kyoto, Japan, September 2016.

[CI46] ***Y. Qamsane*, A. Tajer and A. Philippot**. Distributed Supervisory Control Synthesis For Discrete Manufacturing Systems. 8th IFAC Conference on Manufacturing Modelling, Management & Control (MIM 2016), Troyes, France, June 2016.

[CI45] ***V. Dimanche*, A. Goupil, A. Philippot, B. Riera and A. Urban**. Human factor modeling in railway network. 11th Berlin Human-Machine Systems Workshop (BWMMS 2015), Berlin, Germany, October 2015.

[CI44] **B. Riera, A. Philippot, *R. Coupat*, F. Gellot and D. Annebicque**. A non-intrusive method to make safe existing PLC Program. 9th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes (SAFEPROCESS'15), Paris, France, September 2015.

[CI43] **P. Marangé, A. Philippot, J.-F. Pétin and F. Gellot**. Diagnosability evaluation by model-checking. 9th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes (SAFEPROCESS'15), Paris, France, September 2015.

[CI42] **R. Saddem and A. Philippot**. Utilisation de diagnostiqueurs discrets pour l'établissement de Signatures Temporelles Causales pour le diagnostic des SED. Colloque International sur le Monitoring des Systèmes Industriels 2014 (CIMSI'2014), Marrakech Maroc, December 2014.

[CI41] **R. Saddem and A. Philippot**. Causal Temporal Signature from Diagnoser model for online Diagnosis of Discrete Event Systems. International Conference on Control, Decision and Information Technologies (CoDIT'14), Metz, France, November 2014.

[CI40] ***Y. Qamsane*, A. Tajer and A. Philippot**. Synthesis and implementation of distributed control for a flexible manufacturing system. 2nd World Conference on Complex Systems (WCCS14), Agadir, Morocco, November 2014.

[CI39] **A. Philippot, P. Marangé, F. Gellot, J.-F. Pétin and B. Riera**. Fault Tolerant Control for manufacturing discrete systems by filter and diagnoser interactions. Annual Conference of the Prognostics and Health Management Society (PHM'14), Fort Worth, Texas, USA, October 2014.

[CI38] ***R. Coupat*, M. Meslay, M.-A. Burette, B. Riera, A. Philippot and D. Annebicque**. Standardization and Safety Control Generation for SNCF Systems Engineer. 19th IFAC World Congress 2014 (IFAC WC), Cape Town, South Africa, August 2014.

[CI37] **B. Riera, *R. Coupat*, A. Philippot, F. Gellot and D. Annebicque**. Control design

pattern based on safety logical constraints for Manufacturing Systems: Application to a Palletizer. 12th IFAC - IEEE International Workshop On Discrete Event Systems (WODES'14), Paris, France, 2014.

[CI36] **S. Debernard, F. Gellot, P. Marangé, M. Pacaux, T. Poulain, A. Philippot, B. Riera and J.-F. Pétin**. A support tool for assisting human diagnoses of control errors 10th Berlin Human-Machine Systems Workshop (HMS'13), Berlin, Germany, October 2013.

[CI35] ***R. Coupat*, M. Meslay, M.-A. Burette, A. Philippot, D. Annebicque and B. Riera**. The standardized generation and the robust filtering of the command as tools of optimization of the mental workload of the systems engineer. IFAC Symposium on Analysis, Design, and Evaluation of Human-Machine Systems (HMS 2013), LAS VEGAS, August 2013.

[CI34] **A. Tajer, A. Philippot and V. Carré-Ménétrier**. Synthesis of Optimal controller of Discrete Manufacturing Systems by Liveness Extraction. IEEE International Conference on Complex Systems (ICCS'12), Agadir, Morocco, November 2012.

[CI33] **B. Riera, A. Philippot, D. Annebicque and F. Gellot**. Safe control synthesis based on Boolean constraints for manufacturing systems. 8th Symposium on Fault Detection, Supervision and Safety of Technical Processes (SAFEPROCESS 2012), Mexico City, Mexico, August 2012.

[CI32] **B. Riera, F. Gellot, A. Philippot, B. Vigario and D. Annebicque**. Synthèse de commande sûre de fonctionnement à base de contraintes logiques pour les systèmes manufacturiers. 9th International Conference on Modeling, Optimization & SIMulation (MOSIM'2012), Bordeaux, France, June 2012.

[CI31] **A. Philippot, P. Marangé, V. Carré-Ménétrier and B. Riera**. Implementation of diagnosis approach for Discrete Event Systems. International symposium on Security and Safety of Complex Systems (2SCS'12), Agadir, Morocco, May 2012.

[CI30] **A. Tajer, A. Philippot and V. Carré-Ménétrier**. Design and implementation of decentralized controller for Discrete Manufacturing Systems. International symposium on Security and Safety of Complex Systems (2SCS'12), Agadir, Morocco, May 2012.

[CI29] **B. Riera, D. Annebicque, F. Gellot, A. Philippot and R. Benlorhfar**. Control synthesis based on logical constraints for safe manufacturing systems. 14th IFAC Symposium on INformation COntrol problems in Manufacturing (INCOM 2012), Bucharest, Romania, May 2012.

[CI28] **A. Philippot, V. Carré-Ménétrier and A. Tajer**. Optimal Controller for Manufacturing Systems by Decentralized Approach. European Safety and Reliability Annual Conference (ESREL'11), Troyes, France, September 2011.

[CI27] **A. Philippot and V. Carré-Ménétrier**. Methodology to obtain local discrete diagnosers. 3rd International Workshop on Dependable Control of Discrete Systems

(DCDS'11), pp 49-54, IEEE, Saarbrücken, Germany, June 2011.

[CI26] **A. Philippot**. Survey on Diagnosis of a Pick and Place Benchmark - Special Session on Diagnosis of Discrete Event Systems: Application on a Benchmark. 3rd International Workshop on Dependable Control of Discrete Systems (DCDS'11), pp 27-30, IEEE, Saarbrücken, Germany, June 2011.

[CI25] **A. Philippot and V. Carré-Ménétrier**. Comparison of diagnosis approaches for Discrete Event Systems. International Conference on Industrial Engineering and Systems Management (IESM'11), Metz, France, May 2011.

[CI24] **A. Tajer, A. Philippot and V. Carré-Ménétrier**. Decentralized Implementation Approach of Control Synthesis of Manufacturing Systems. 2nd International Conference on Multimedia Computing and Systems (ICMCS'11), IEEE, Ouarzazate, Morocco, April 2011.

[CI23] **A. Philippot, A. Tajer and V. Carré-Ménétrier**. Elaboration of Distributed Optimal Controller for Manufacturing Systems through Synthesis Approach. International Conference on Communication, Computing and Control Applications (CCCA'11), IEEE, Hammamet, Tunisia, March 2011.

[CI22] **A. Philippot and A. Tajer**. Conception de la commande sûre des systèmes manufacturiers à base du Graphe Equivalent du GRAFCET. 2nd édition du colloque international sur les Systèmes Industriels et Logistiques (SIL'10), Marrakech, Maroc, October 2010.

[CI21] ***M. Hemour*, A. Philippot, N. Messai, D. Caligny and B. Riera**. Human design applied to operating light rail studies. 11th IFAC/IFIP/IFORS/IEA Symposium on Analysis, Design, and Evaluation of Human-Machine Systems (HMS'2010), Valenciennes, France, September 2010.

[CI20] **A. Philippot and A. Tajer**. From GRAFCET to Equivalent Graph for Synthesis Control of Discrete Events Systems. 18th Mediterranean Conference on Control and Automation (MED10), pp 683-688, IEEE, Marrakech, Morocco, June 2010.

[CI19] ***M. Hemour*, N. Messai, A. Philippot, D. Caligny and B. Riera**. Human adapted design for operating metro studies. 28th European Annual Conference on Human Decision-Making and Manual Control (EAM09), Reims, France, September 2009.

[CI18] ***N. Malki*, A. Philippot, M. Sayed-Mouchaweh and V. Carré-Ménétrier**. Independent codiagnosability of discrete event systems using component based-approach: from modelling to diagnosis. 28th European Annual Conference on Human Decision-Making and Manual Control (EAM09), Reims, France, September 2009.

[CI17] **A. Philippot, M. Sayed-Mouchaweh and V. Carré-Ménétrier**. Distributed Modeling Approach of discrete manufacturing systems by Parts of Plant. 10th European Control Conference 2009 (ECC'09), Budapest, Hungary, August 2009.

[CI16] **A. Philippot, M. Sayed-Mouchaweh and V. Carré-Ménétrier**. Candidates' generation for diagnosis of discrete manufacturing systems. 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes (SAFEPROCESS'09), Barcelona, Spain, July 2009.

[CI15] **A. Philippot, M. Sayed-Mouchaweh and V. Carré-Ménétrier**. Modelling of a discrete manufacturing system by Parts of plant. 13th IFAC Symposium on INformation COntrol problems in Manufacturing (INCOM'09), Moscow, Russia, June 2009.

[CI14] **A. Philippot, M. Sayed-Mouchaweh, V. Carré-Ménétrier and B. Riera**. Discrete Event Model-Based Approach for Fault Detection and Isolation of Manufacturing Systems. 2nd IFAC Workshop on Dependable Control of Discrete Systems (DCDS'09), Bari - Italy, June 2009.

[CI13] **M. Sayed-Mouchaweh, A. Philippot, V. Carré-Ménétrier and B. Riera**. Fault Diagnosis of Discrete Event Systems Using Components Fault-Free models. 20th International Workshop on Principles of Diagnosis (DX'09), Stockholm, Sweden, June 2009.

[CI12] **A. Tajer and A. Philippot**. Evolution of Plant Modeling Approaches for Discrete Manufacturing Systems. International Conference on Multimedia Computing and Systems (ICMCS 09), pp 345-350, IEEE, Ouarzazate, Morocco, April 2009. ISBN: 978-1-4244-3756-6.

[CI11] **A. Philippot, M. Sayed-Mouchaweh and V. Carré-Ménétrier**. Nouveaux Modèles De Partie Opérative des Systemes Manufacturiers. 5ème Conférence Internationale Francophone d'Automatique (CIFA08), Bucarest-Roumanie, September 2008.

[CI10] **A. Philippot, M. Sayed-Mouchaweh and V. Carré-Ménétrier**. Unconditional Decentralized Structure for the fault diagnosis of Discrete Event Systems. 1st IFAC Workshop on Dependable Control of Discrete-event Systems (DCDS'07), Cachan, France, June 2007.

[CI9] **M. Sayed-Mouchaweh, A. Philippot and V. Carré-Ménétrier**. Decentralized Approach for fault diagnosis of Discrete Event Systems. IFAC 4th International Conference on Informatics in Control, Automation and Robotics (ICINCO'07), Angers, France, May 2007.

[CI8] **A. Philippot, M. Sayed-Mouchaweh, V. Carré-Ménétrier and B. Riera**. Decentralized approach to diagnose manufacturing systems. IMACS Multiconference on Computational Engineering in Systems Applications (CESA06), vol. 1, pp 912-918, IEEE, Beijing, China, October 2006.

[CI7] **M. Sayed-Mouchaweh, A. Philippot, V. Carré-Ménétrier and B. Riera**. Timed-Event-State-Based Diagnoser for Manufacturing systems. 7th IFIP International Conference on Information Technology for Balanced Automation Systems in

Manufacturing and Services (BASYS'06), Niagara Falls, Ontario, Canada, September 2006.

[CI6] **M. Sayed-Mouchaweh, A. Philippot and V. Carré-Ménétrier**. Detectability and Diagnosability of Discrete Event Systems: Application on manufacturing systems. 2nd ICINCO/IFAC International Conference Informatics in Control, Automation and Robotic (ICINCO'05), pp 149-154, Barcelona, Spain, September 2005.

[CI5] **A. Philippot, M. Sayed-Mouchaweh and V. Carré-Ménétrier**. Multi-models approach for the diagnosis of Discrete Events Systems. 17th IMACS World Congress (IMACS'05), Actes sur CD-ROM, Paris, France, July 2005.

[CI4] **A. Tajer, A. Philippot, F. Gellot and V. Carré-Ménétrier**. Démarche formelle de synthèse de synthèse d'une commande sûre à partir d'une spécification GRAFCET. Conférence Internationale Francophone de l'Automatique (CIFA'04), Douz, Tunisie, November 2004.

[CI3] **A. Philippot, A. Tajer, F. Gellot and V. Carré-Ménétrier**. Méthodologie de modélisation dans le cadre de la synthèse formelle des SED. Conférence Internationale Francophone de l'Automatique (CIFA'04), Douz, Tunisie, November 2004.

[CI2] **M. Sayed-Mouchaweh, A. Philippot, S. Triki and B. Riera**. Separated approach for "active" monitoring of discrete event systems. IFAC 7th Workshop on Discrete Event Systems (WODES'04), pp 381-386, Reims, France, September 2004.

[CI1] **A. Philippot, A. Tajer, F. Gellot and V. Carré-Ménétrier**. On-line synthesis approach based on a structured plant modelling. IFAC 7th Workshop on Discrete Event Systems (WODES'04), pp 381-386, Reims, France, September 2004.

CONFERENCES NATIONALES :

[CN22] ***M. Chankate*, A. Philippot, P. Marangé and V. Carré-Ménétrier**. Conception d'un Système de Vérification de la Diagnosticabilité par Model-Checking à partir du modèle du système. 12th International Conference on Modeling, Optimization & SIMulation (MOSIM'2018), Toulouse, France, June 2018.

[CN21] ***M. Niang*, A. Philippot, F. Gellot, R. Coupat, B. Riera, and S. Lefebvre**. Vérification formelle des programmes automates de la SNCF par Model-Checking. 11ème Colloque sur la Modélisation des Systèmes Réactifs (MSR 2017), Marseille, France, novembre 2017.

[CN20] **B. Riera, *R. Pichard*, A. Philippot, R. Saddem, F. Gellot, D. Annebicque et F. Emprin**. HOME I/O et FACTORY I/O : 2 logiciels innovants de simulation de PO pour la formation à l'automatique. 12ème Colloque Enseignement des Technologies et des Sciences de l'Information et des Systèmes (CETSIS 2017), Le Mans, France, mai 2017.

[CN19] ***I. Tahiri*, A. Tajer, *Y. Qamsane* et A. Philippot**. Contribution à la commande des Systèmes à Evénements Discrets Temporisés (SEDTS). 11ème Conférence Francophone de Modélisation, Optimisation et Simulation (MOSIM'16), Montréal, Québec, Canada, août 2016.

[CN18] **B. Riera, A. Philippot, D. Annebicque et F. Gellot**. La commande par contraintes logiques de sécurité : principe, applications et mise en œuvre. 10ème Colloque sur la Modélisation des Systèmes Réactifs (MSR 2015), Nancy, France, novembre 2015.

[CN17] ***Y. Qamsane*, A. Tajer et A. Philippot**. Synthèse de contrôle distribué pour un système manufacturier. 10ème Conférence Francophone de Modélisation, Optimisation et Simulation (MOSIM'14), Nancy, France, novembre 2014.

[CN16] **P. Marangé, A. Philippot, J.-F. Pétin et F. Gellot**. Vérification de la diagnosticabilité par Model-Checking 10ème Conférence Francophone de Modélisation, Optimisation et Simulation (MOSIM'14), Nancy, France, novembre 2014.

[CN15] **B. Riera, *R. Coupat*, D. Annebicque, A. Philippot et F. Gellot**. Control synthesis based on safety Boolean guards for manufacturing systems: application to a sorting system. 10ème Conférence Francophone de Modélisation, Optimisation et Simulation (MOSIM'14), Nancy, France, novembre 2014.

[CN14] **A. Philippot, B. Riera, F. Gellot, D. Annebicque, *R. Coupat* et E. Pierrel**. Initiation à la qualimétrie de code d'automate programmable industriel. 11ème Colloque Enseignement des Technologies et des Sciences de l'Information et des Systèmes (CETSIS 2014), Besançon, octobre 2014.

[CN13] **A. Philippot et A. Philippot**. Rush Hour : Introduction au Model-Checking à travers le jeu. 11ème Colloque Enseignement des Technologies et des Sciences de l'Information et des Systèmes (CETSIS 2014), Besançon, octobre 2014.

[CN12] ***R. Coupat*, M.-A. Burette, A. Philippot, D. Annebicque et B. Riera**. Synthèse de la commande sûre de fonctionnement et génération automatique de code API pour les systèmes distribués reconfigurables. JDMACS 2013, Strasbourg, juillet 2013.

[CN11] **B. Riera, B. Vigario, A. Philippot, D. Annebicque et F. Gellot**. Simulateur 3D interactif de Parties Opératives et Synthèse sûre de la commande des systèmes manufacturiers. 4ème Journées des Démonstrateurs en Automatique Section automatique du club EEA, Angers, France, juin 2013.

[CN10] **D. Annebicque, A. Philippot, F. Gellot et B. Riera**. Un interpréteur de GRAFCET pour l'enseignement et la recherche. 10ème Colloque Enseignement des Technologies et des Sciences de l'Information et des Systèmes (CETSIS 2013), Caen, mars 2013.

[CN9] **F. Gellot, A. Philippot, D. Annebicque et B. Riera**. Commande robuste et sûre de fonctionnement des systèmes manufacturiers. 14ème conférence ROADEF de la

société Française de Recherche Opérationnelle et Aide à la Décision (ROADEF'13), Troyes, France, février 2013.

[CN8] **B. Riera, F. Gellot, D. Annebicque, B. Vigario et A. Philippot**. Synthèse de commande sûre de fonctionnement à base de contraintes logiques pour les systèmes manufacturiers. 9ème Conférence Internationale de Modélisation, Optimisation et SIMulation Performance, interopérabilité et sécurité pour le développement durable (MOSIM'2012), Bordeaux, France, juin 2012.

[CN7] **A. Philippot, A. Tajer et V. Carré-Ménétrier**. Approche Décentralisée pour le Diagnostic des Systèmes Manufacturiers. Conférence Méditerranéenne sur l'ingénierie sûre des systèmes complexes (MISC'11), Agadir, Maroc, mai 2011.

[CN6] **A. Tajer, A. Philippot et V. Carré-Ménétrier**. Elaboration d'une Commande des Systèmes Manufacturiers utilisant une Approche Décentralisée. Conférence Méditerranéenne sur l'ingénierie sûre des systèmes complexes (MISC'11), Agadir, Maroc, mai 2011.

[CN5] ***M. Hemour*, A. Philippot, N. Messai, D. Caligny et B. Riera**. Amélioration de l'intégration du facteur humain dans les études d'exploitation ferroviaire. 6ème Conférence Internationale Francophone d'Automatique (CIFA 2010), Nancy, France, juin 2010.

[CN4] **A. Philippot, M. Sayed-Mouchaweh et V. Carré-Ménétrier**. Démarche globale de diagnostic décentralisé des SED : Application à un système de transfert de pièces. 6ème Colloque Francophone sur la Modélisation des Systèmes Réactifs (MSR'07), Lyon, France, octobre 2007.

[CN3] **A. Philippot, M. Sayed-Mouchaweh et V. Carré-Ménétrier**. Diagnostic des SED par modélisation multi-outils. Journées Doctorales MACS (JDMACS 2005), Lyon, France, septembre 2005.

[CN2] **A. Philippot, A. Tajer, F. Gellot et V. Carré-Ménétrier**. Synthèse de la commande spécifiée en GRAFCET: application à un préhenseur pneumatique. Colloque Francophone sur la modélisation des systèmes réactifs (MSR'03), pp 61-75, Metz, octobre 2003.

[CN1] **A. Tajer, A. Philippot, F. Gellot et V. Carré-Ménétrier**. Contribution à l'amélioration de la praticabilité des approches formelles de synthèse de commande. Journées Doctorales d'Automatique (JDA'03), pp 239-244, Valenciennes, France, juin 2003.

RAPPORTS ET COMMUNICATIONS SANS ACTE :

[CSA14] ***R. Pichard*, M. Combacau, A. Philippot, R. Saddem and B. Riera**. SEDMA - un outil pour la Modélisation, l'Analyse et la génération automatique de programme pour les SED. 11ème Colloque sur la Modélisation des Systèmes Réactifs (MSR

2017), Session Poster/Démonstrateur, Marseille, France, novembre 2017.

[CSA13] **F. Vanderhaegen, A. Philippot**. Spécification des Besoins. Livrable 1 du Projet inter-régional SUCRé (Sûreté de fonctionnement et résilience pour la gestion et le contrôle coopératif des systèmes sociotechniques) entre les régions Nord-Pas-de-Calais et Champagne-Ardenne - Coopération Homme(s)-Robot(s) en milieu hostile (2015-2017), Septembre 2015.

[CSA12] ***R. Coupat*, B. Riera, *M. Niang*, F. Gellot, M. Meslay, A. Philippot, H. Caron and D. Annebicque**. Automatic generation of safe PLC programs. 9th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes (SAFEPROCESS'15), Paris, France, September 2015.

[CSA11] ***H. Dimanche*, A. Philippot**. Contraintes de sécurité et diagnostic des SED. Rapport de Stage Master Recherche. Septembre 2013.

[CSA10] ***H. Serradj*, A. Philippot**. Mise en place d'une approche de diagnostic sur systèmes manufacturiers à travers une méthode de classification des défaillances. Rapport de Stage Recherche. Juillet 2013.

[CSA9] ***E. Tieche*, F. Gellot, A. Philippot, D. Annebicque et B. Riera**. Supervision de l'atelier flexible – Programme de démonstration. Sept. 2013.

[CSA8] ***E. Tieche*, F. Gellot, A. Philippot, D. Annebicque et B. Riera**. Supervision de l'atelier flexible – Programme avec filtre logique robuste aux erreurs de commande. Sept. 2013.

[CSA7] ***E. Tieche*, F. Gellot, A. Philippot, D. Annebicque et B. Riera**. Application des contraintes de sécurité sur la cellule flexible CellFlex. Janv. 2013.

[CSA6] **P. Marangé, J.-F. Pétin, D. Gouyon, B. Riera, F. Gellot, A. Philippot, S. Debernard, M. Pacaux et T. Poulain**. Approche de Détection et d'Explication de Commande par filtrage robuste Projet ADEXEC. Workshop Surveillance, Sûreté et Sécurité des Grands Systèmes (3SGS'11), Valenciennes, octobre 2011.

[CSA5] **A. Philippot**. Elaboration de diagnostiqueurs locaux à partir de modèles distribués de Partie Opérative. Ecole des JDMACS - 3èmes Journées Doctorales / Journées Nationales MACS (JD-JN-MACS), Angers, mars 2009.

[CSA4] **A. Philippot, M. Sayed-Mouchaweh et V. Carré-Ménétrier**. Elaboration de diagnostiqueurs locaux à partir de modèles distribués de Partie Opérative. Présentation au GT INCOS lors des Journées du Pôle STP du GDR MACS, Metz, France, novembre 2008.

[CSA3] **A. Philippot**. Evaluation du langage SFC+ de l'atelier LCM Studio en vue de la simulation de commande des systèmes manufacturiers. Rapport Post Doctoral - LURPA. 29 pages, 2007.

[CSA2] **A. Philippot**. Documentation LCM Studio / DELMIA Automation. Rapport interne

au LURPA (Réalisé sous SP2). 19 pages, 2007.

[CSA1] **A. Philippot, M. Sayed-Mouchaweh et V. Carré-Ménétrier**. Une approche multi-outils décentralisée et modulaire pour le diagnostic des Systèmes à Evénements Discrets. Présentation au GT INCOS lors des Journées du Pôle STP du GDR MACS, Clermont-Ferrand, France, mars 2005.

[CSA0] **A. Philippot**. Contribution au diagnostic décentralisé des SED : application aux systèmes manufacturiers. Thesis, 2006.

6. Conclusion du chapitre

Ce premier chapitre présente la synthèse de l'ensemble de mes activités pédagogiques, de recherche, de mes différentes responsabilités et participations dans la communauté depuis ma nomination en tant que maître de conférences en 2007.

En tant qu'Enseignant/Chercheur, j'accorde une grande importance au devoir de transmission envers les étudiants. Cet intérêt s'exprime par le fait que l'enseignement m'est apparu comme une vocation et cela depuis l'adolescence. Ainsi, après des études plutôt technologiques et professionnelles, lorsqu'il m'a été possible d'intégrer un DEA dans un objectif d'effectuer à terme de l'enseignement, cela m'est apparu comme une évidence. Durant cette année de Bac+5, j'ai également pu découvrir la recherche en laboratoire. Cette possibilité de « switcher » entre deux activités et même de les faire parfois cohabiter m'a séduite. L'implication dans la pédagogie est sûrement excessive à la vue du nombre d'heures effectuées dans mon service. Néanmoins, il est difficile de refuser d'accompagner des étudiants dans leur parcours professionnel lorsque ceux-ci sont moteurs et demandeurs (à l'image du suivi que je réalise lors des concours Robafis par exemple).

Concernant la partie recherche, l'activité n'a jamais été discontinue et couvre les thèmes de la commande et du diagnostic des Systèmes à Evénements Discrets à base de modèles. Il s'agit d'une recherche locale mais aussi nationale avec la collaboration autour de projets avec des collègues du LAMIH de Valenciennes et du CRAN de l'Université de Lorraine. Au niveau international, les collaborations y sont encore moindre mais tout de même présentes avec la participation au GDR International HAMASYTI (*International Research Network on HumAn-MAchine SYstems in Transportation and Industry*) et la représentation au comité technique "TC-3.1. Computers for Control" de l'IFAC "International Federation of Automatic Control". Le laboratoire a également accueilli le Pr. Martin Fabian de l'Université de

Chalmers afin que l'équipe CDSED puisse échanger avec lui sur nos travaux autour de la théorie de la supervision.

Mon activité de recherche est couverte par des publications de qualité dans des conférences internationales (2 *Best Student Paper Award* pour Y. Qamsane et I. Tahiri). La publication en revue internationale a démarré tardivement mais est croissante depuis 2016. Enfin, malgré une participation active à plusieurs projets de recherche, je ne me suis jamais encore retrouvé porteur de l'un deux. Ce point devra être corrigé à l'avenir.

En termes de responsabilité et de rayonnement, je m'occupe de formations et d'Unités d'Enseignement (UE) au sein de l'URCA et je pense avoir la reconnaissance de mes collègues (soutien par des HRS). Je suis également actif dans les sociétés savantes telles que le Club EEA ou le GT SED. Elu au CNU 61ème section, j'y participe régulièrement (sessions qualifications, promotions suivi de carrière) et j'ai également appris beaucoup sur le métier en échangeant avec les collègues des différentes disciplines. Enfin, l'industrie du futur étant un sujet d'actualité au sein de nos formations et dans les activités du CReSTIC, j'ai récemment été nommé Chargé de Mission « Industrie du Futur » par le Président de l'URCA. Cette nomination est une reconnaissance de mon travail au sein de l'université. Elle me permet notamment d'échanger avec les industriels et chercheurs au niveau local et national.

Le deuxième chapitre de ce mémoire reprend plus en détails mes principaux travaux sur la modélisation pour la commande et le diagnostic des Systèmes à Evénements Discrets.

Chapitre 2 : Conception d'approches de Commande et de Diagnostic des SED

1. Introduction et problématique

A l'origine, l'automatique s'est intéressée à la commande des systèmes dynamiques décrits par des équations différentielles ou des dérivées partielles en temps continu ou discret afin d'identifier des phénomènes physiques. Toutefois, lorsque la dynamique est représentée par le déclenchement d'événements ponctuels, cette représentation est beaucoup moins adaptée. On parle alors de Systèmes à Evènements Discrets (SED), et ils apparaissent de façon naturelle dans la modélisation des systèmes informatiques et embarqués, des réseaux de télécommunication, des réseaux de transport ou des systèmes de production. Un SED est un système qui satisfait les deux propriétés suivantes : l'espace d'états est discret et la transition d'états est déclenchée par un événement. Les Systèmes Automatisés de Production (SAP) constituent un champ d'applications privilégié des travaux de recherche en SED au sein du CReSTIC. La complexité croissante des SAP en termes de quantité, de besoins en communication, de diversité des composants, etc., entraine une augmentation des exigences des utilisateurs concernant la sûreté de fonctionnement des contrôleurs et le diagnostic des installations. A l'ère de l'Industrie du futur, pouvoir garantir de manière formelle la sécurité des systèmes automatisés et augmenter leur productivité et flexibilité est un défi scientifique majeur.

Inscrit dans l'équipe CDSED (Commande et Diagnostic des SED), mes travaux se positionnent sur l'étude et l'analyse des SED avec un objectif de commande et de diagnostic. Il s'agit de contribuer à l'apport d'outils méthodologiques et théoriques permettant d'améliorer les performances du système global composé d'une Partie Opérative (PO) et d'une Partie Commande (PC). L'originalité des travaux est de proposer des algorithmes de commande et de diagnostic implémentables dans des calculateurs Industriels (Automate Programmable Industriel par exemple). Cela nécessite entre autres de proposer des solutions novatrices n'entrainant pas d'explosion combinatoire. Il est également nécessaire de prendre

en compte la composante humaine dans la conception des outils et méthodes développées. En effet, ces développements visent une amélioration de la performance du système global dans lequel l'homme reste un maillon indispensable vu comme une valeur ajoutée. Toutefois, ces artefacts parce qu'ils modifient la place, le rôle et les tâches de l'opérateur, qu'il soit automaticien, opérateur de supervision ou responsable de la maintenance doivent être conçus dans une approche de coopération Homme-Machine.

Ce chapitre est décomposé en 2 sections reprenant l'ensemble de mes recherches autour de la Commande et du Diagnostic (Figure 2-1). On y retrouve la place des projets selon les thématiques et une partie de mes encadrements. Par ailleurs, les SED recouvrent de nombreux outils permettant la modélisation à différents niveaux d'abstraction. La représentation des SED dépend essentiellement de la granularité de modélisation et de l'objectif recherché. Les formalismes utilisés sont soit à base d'algèbres (algèbre de Boole pour les expressions de contraintes logiques), soit à base d'états-transitions (automates à états finis, le GRAFCET, les réseaux de Petri, StateChart, …).

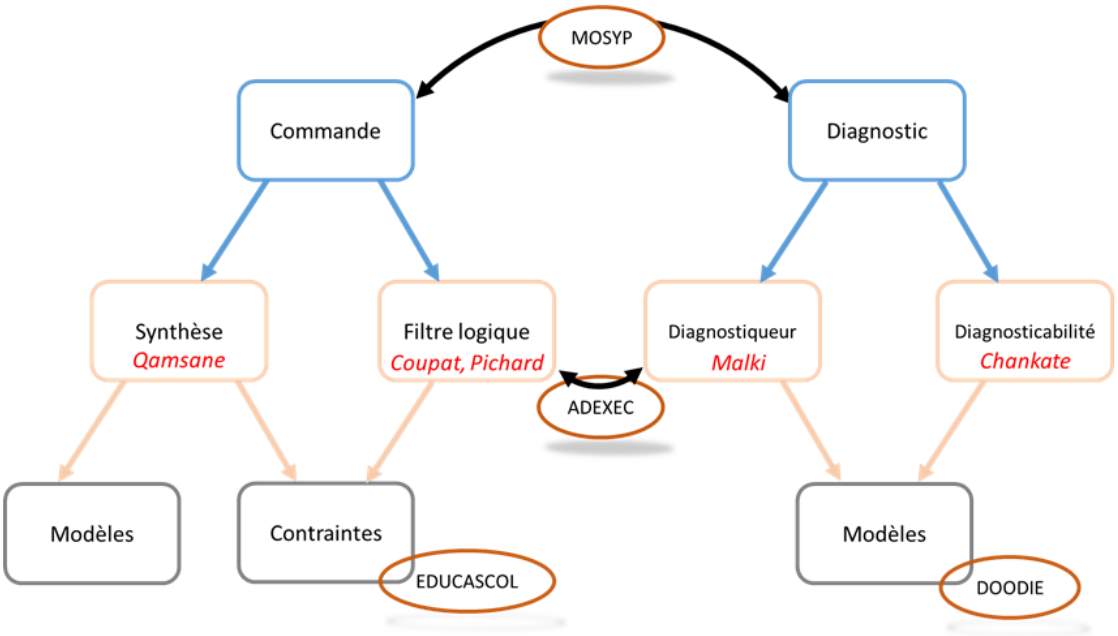

Figure 2-1 : Travaux d'études depuis 2003

2. Commande des Systèmes à Evénements Discrets

L'accroissement de la demande d'automatisation et la complexité des systèmes de production ont conduit à l'utilisation de méthodes formelles. La littérature scientifique propose deux approches pour répondre à cet objectif : i) la théorie de contrôle par supervision (*Supervisory Control Theory* - SCT) (Ramadge and Wonham, 1989) et la Vérification et Validation (V&V) de modèle. Que ce soit l'une ou l'autre des approches, les difficultés de calcul et plus généralement le problème d'explosion combinatoire s'oppose à leur application dans le monde industriel.

Les travaux proposés sur la synthèse sûre de fonctionnement des contrôleurs logiques consistent en la recherche d'une commande dite « optimale » à travers l'utilisation de modèles décentralisés. En effet, l'approche originale se basant sur la SCT classique demande une étape de modélisation de la partie opérative (PO) du système. Cette étape est très souvent « gourmande » au niveau de la taille des modèles et représente un frein à son applicabilité. La proposition consiste à ne pas disposer d'un modèle global de la PO, mais de plusieurs modèles locaux sur lesquels il est possible d'appliquer des contraintes de sécurité locales.

Par ailleurs, une méthode par filtre à base de fonctions logiques (appelées contraintes), dont les propriétés de sécurité et vivacité sont vérifiées hors ligne par *model-checking*, implémentées à la fin du programme de commande du contrôleur a été proposée au sein de l'équipe. Cette approche originale permet soit de fiabiliser des programmes Automates Programmables Industriels (API) existants, soit de développer des contrôleurs sûrs de fonctionnement originaux où les aspects fonctionnels et sécuritaires sont séparés.

2.1. Rappel sur la Théorie de Supervision (*Supervisory Control Theory*)

La théorie de la commande par supervision (Ramadge and Wonham, 1989) a pour principal objectif d'établir pour un SED des concepts et notions analogues à ceux de l'automatique continue tels que la commandabilité et l'observabilité. Elle emploie pour ceci des techniques issues du domaine de l'informatique telles que les langages formels et les automates. Un superviseur est un SED qui, ayant l'habilité d'interdire l'occurrence de certains événements, permet de restreindre le fonctionnement possible du procédé. Étant donnés un

procédé et un ensemble de spécifications de commande, cette théorie permet une synthèse systématique de superviseurs.

La notion de commandabilité garantit par construction que le fonctionnement du procédé, couplé à son superviseur, respecte les spécifications de commande prédéfinies (Kumar, 1991). En outre, on peut assurer que le fonctionnement ainsi obtenu est non bloquant. Il est également souhaitable que la commande résultante soit optimale. Dans ce contexte, l'optimalité signifie que le comportement commandé est aussi permissif que possible.

Pour commander un procédé dans le cadre de la SCT, certains événements sont interdits ou autorisés en temps voulu (Figure 2-2). Le procédé G est couplé au superviseur S. Les entrées du superviseur sont les sorties du procédé, et les entrées du procédé sont les sorties du superviseur. Le procédé G est un SED qui génère des événements observables de manière instantanée, spontanée, et asynchrone. Le superviseur S est un SED qui évolue conformément aux événements générés par le procédé. Il peut modifier le fonctionnement du procédé en observant la séquence des événements générés par celui-ci et après chaque événement généré, il autorise ou interdit l'occurrence des événements commandables de manière à satisfaire les spécifications.

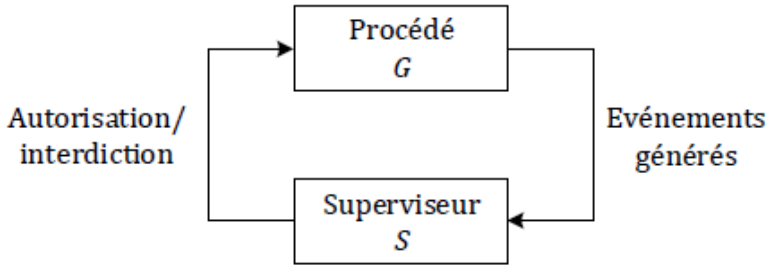

Figure 2-2 : Principe de la SCT

Le procédé est modélisé par un automate G = (Q, Σ, δ, Qm, q_0) avec :

- Q est un ensemble fini d'états ;
- $\Sigma = \Sigma_c \cup \Sigma_{uc}$ est l'ensemble des événements (alphabet) avec Σ_c l'ensemble des événements contrôlables et Σ_{uc} l'ensemble des événements non contrôlables ;
- $\delta : Q \times \Sigma \rightarrow Q$ est la fonction de transition d'états ;
- $Q_m \subseteq Q$ est l'ensemble des états marqués ;
- q_0 est l'état initial.

Il génère le langage formel L(G) représentant l'ensemble des séquences d'événements qui

peuvent être générées par G. Ce langage est nécessairement préfixe-clos du fait que le langage accepté par l'automate du procédé est régulier. Ce procédé représente l'ensemble des comportements que peut potentiellement réaliser le système étudié. Le procédé est couplé à un superviseur, représenté lui aussi par un automate S évoluant conformément aux événements issus du procédé. Le superviseur agit sur l'évolution du procédé par l'intermédiaire de lois de commande qui définissent l'ensemble des événements autorisés et interdits à partir de l'état où se trouve le procédé. Le couplage superviseur/procédé (S/G) doit assurer le respect des spécifications données par le cahier des charges.

De nombreux ouvrages traitent de la SCT d'un point de vue théorique (Cassandras and Lafortune, 2008; Wonham, 2015). L'application aux systèmes réels reste par contre encore complexe à obtenir. L'adoption d'approches centralisées induisant une explosion de l'espace d'état lors des phases de modélisation des systèmes de tailles importantes en est notamment responsable. D'autre part, l'évolution rapide du contexte économique dans lequel opèrent les industriels, impose l'adoption de systèmes flexibles. La notion de flexibilité reflète l'aptitude à rester opérationnel dans des situations changeantes. Disposer d'un système flexible de production requiert de prédire sa capacité d'adaptation. Dans le cas de la synthèse d'un contrôleur monolithique, il est difficile de s'adapter aux changements de la demande. Même quand un changement ne concerne qu'un seul composant du système global, il faudrait mettre à jour tout le contrôleur monolithique. La conception de contrôleur doit donc prendre en considération l'aspect flexibilité d'implémentation de façon à ce qu'un changement structurel du système n'entraîne pas la mise à jour du contrôleur monolithique, mais que d'un petit nombre de contrôleurs locaux concernés. La plupart des complications dans le domaine des SED souffrent de ces mêmes problèmes de complexité de calcul et de flexibilité d'implémentation quel que soit le cadre particulier ou l'approche adoptée. Pour résoudre ces problèmes, des travaux utilisent des stratégies de décomposition des différents modèles pour limiter le nombre d'états à traiter, ce qui conduit à introduire les aspects modulaires, hiérarchiques, décentralisés et distribués dans la théorie de commande par supervision. (Zaytoon and Riera, 2017) ont proposé un tour d'horizon des approches de synthèse et d'implémentation pour les contrôleurs logiques.

Depuis le début des années 2000, la synthèse de commande pour les SAP a toujours été un thème de recherche pour l'équipe SED du laboratoire. Ce sujet a évolué au cours du temps

dans ses architectures, ses algorithmes et ses modèles. Je propose de les reprendre succinctement dans la section suivante.

2.2. Evolution des approches proposées pour la Synthèse de commande basées sur la SCT

2.2.1. Approche initiale

Le principe de la démarche de synthèse d'une commande sûre de fonctionnement initialement proposée par le laboratoire repose sur les concepts de la Figure 2-3. L'objectif est de fournir, pour la commande d'un procédé donné spécifiée par GRAFCET (IEC 60848, 2002), une implantation sur Automate Programmable Industriel (API) qui soit réactive, déterministe et sans blocage. Cette implantation doit alors avoir le comportement le plus large possible par rapport à la commande et un ensemble de contraintes de sécurité et de vivacité (Carré-Ménétrier and Zaytoon, 2002). La démarche s'articule autour d'une étape de « préparation » à la génération de la commande, d'une étape de génération de la commande proprement dite et d'une étape d'exécution de la commande.

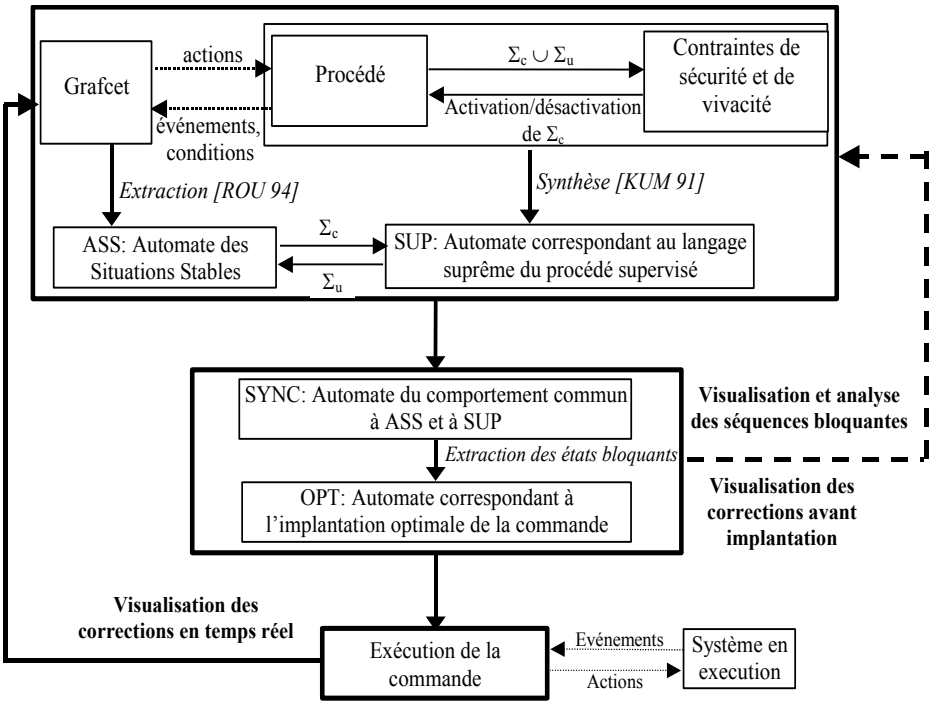

Figure 2-3 : Démarche de synthèse formelle à partir de spécifications GRAFCET

Pour la première étape, le concepteur doit modéliser la commande sous forme de GRAFCET, le comportement de la partie opérative et des contraintes de sécurité et/ou de vivacité sous forme d'automates conformes à ceux utilisés dans la théorie de supervision. A partir de ces modèles, sont élaborés deux automates : l'automate des situations stables *ASS* (Roussel, 1994) représentant l'automate de commande qui est déterministe et réactif, et l'automate du superviseur *SUP* décrivant le plus large comportement commandable admissible du procédé par rapport aux contraintes spécifiées (Kumar, 1991). Dans la seconde étape, les actions du GRAFCET lorsqu'elles ne sont pas admissibles par *SUP*, sont filtrées grâce à une intersection particulière des automates *SUP* et *ASS*. Il s'agit ensuite de retirer les évolutions bloquantes et non atteignables pour générer *OPT* le graphe d'états correspondant à la commande optimale. Cette démarche de synthèse étant sensible aux erreurs de modélisation, le concepteur est alors intégré dans la boucle d'élaboration de la commande, en lui proposant de visualiser la ou les séquences bloquantes, lui permettant ainsi d'agir sur les raisons du blocage en modifiant par exemple la commande, en relâchant des contraintes ou en affinant le modèle de la partie opérative. De nouvelles itérations de la synthèse sont alors effectuées à partir des modèles corrigés pour générer en finalité un modèle de commande correcte (Tajer *et al.*, 2003). Le concepteur peut également visualiser les corrections apportées par la démarche avant l'implantation de la commande. Ces corrections portent sur l'interdiction des actions du GRAFCET non autorisées dans l'état courant du graphe de commande connecté au procédé. Dans la dernière étape de la démarche, la commande optimale implantée est exécutée.

L'applicabilité de cette démarche a été démontrée sur plusieurs exemples. Cependant, pour être réellement praticable sur des systèmes réels complexes, elle se heurte à des problèmes liés à la difficulté de modélisation du comportement du procédé et des contraintes ainsi qu'à l'explosion combinatoire inhérente aux modèles utilisés. Pour comprendre l'intérêt d'une méthodologie de modélisation, une première approche de la modélisation de la partie opérative a été développée. Le même principe est applicable sur les contraintes.

2.2.2. Modélisation théorique structurée de la Partie Opérative

La description précise du comportement de la partie opérative est une opération complexe

car les évolutions d'un système physique sont de nature asynchrone et non déterministe. Pour contourner les difficultés d'une modélisation globale et souvent intuitive, une démarche modulaire permettant d'exprimer des causalités simples entre les Eléments de la Partie Opérative (EPO) a été proposée. La modélisation se base sur des règles définissant les interactions entre les événements commandables et les événements non commandables et de relations précisant les liens entre les événements non commandables. Il s'agit dans le premier cas de règles dites d'occurrence et dans le second cas, de relations de précédence. Ces règles définies par l'utilisateur sont ensuite traduites en automates compatibles.

Avant de déterminer les règles d'occurrence des événements commandables, il faut commencer par fixer le contexte c'est-à-dire les conditions initiales du système. Il s'agit ensuite de recenser tous les événements liés à l'élément de partie opérative que l'on cherche à modéliser puis de définir sous forme de règles, l'influence de l'activation et de la désactivation des événements commandables sur les événements non commandables. Par conséquent, chaque règle va s'exprimer selon le principe simple d'une « cause/conséquence », la cause est liée à l'événement commandable et la conséquence porte sur l'événement non commandable. Pour chaque règle d'occurrence ayant la même cause, il faut ensuite établir sous forme de relations de précédence, la chronologie liant les événements non commandables conséquents.

La construction de l'automate de la partie opérative s'appuie sur les informations précédemment acquises et s'effectue selon les quatre étapes suivantes :

1. Construire l'automate à 2^n états (n étant le nombre d'événements commandables) décrivant toutes les évolutions possibles des événements commandables à partir des conditions initiales données.
2. Compléter l'automate obtenu précédemment, par les événements non commandables résultant des règles d'occurrence,
3. Construire un automate dit de précédence à partir des relations de précédence et de la situation initiale des événements non commandables,
4. Faire le produit croisé synchrone de l'automate issu des règles d'occurrence et de l'automate de précédence.

Cette méthode de modélisation s'effectue pour chaque partie du système. Le modèle global de partie opérative s'obtient alors par composition asynchrone de tous les modèles.

Une illustration simple peut être donnée avec la construction d'un modèle de Vérin Double Effet (VDE) piloté par un distributeur pneumatique 5/2 bistable muni de 2 détecteurs de fin course *a* et *b* (Figure 2-48). Le vérin est commandé en sortie par l'ordre *SO* et en rentrée par l'ordre *RE*.

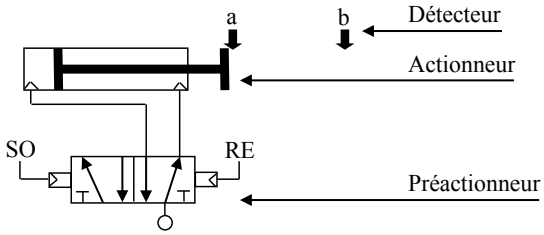

Figure 2-4 : Vérin double effet piloté par un distributeur 5/2 bistable

Le premier modèle intuitif obtenu à l'époque est illustré en Figure 2-5. A comparer avec celui obtenu par modélisation théorique structurée en Figure 2-6d issu des automates des Figure 2-6a, b et c interprétés du tableau d'interactions et de relations (Tableau 2-1).

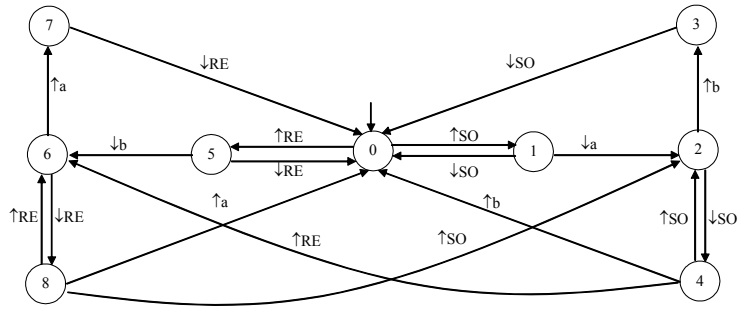

Figure 2-5 : Modèle intuitif du vérin

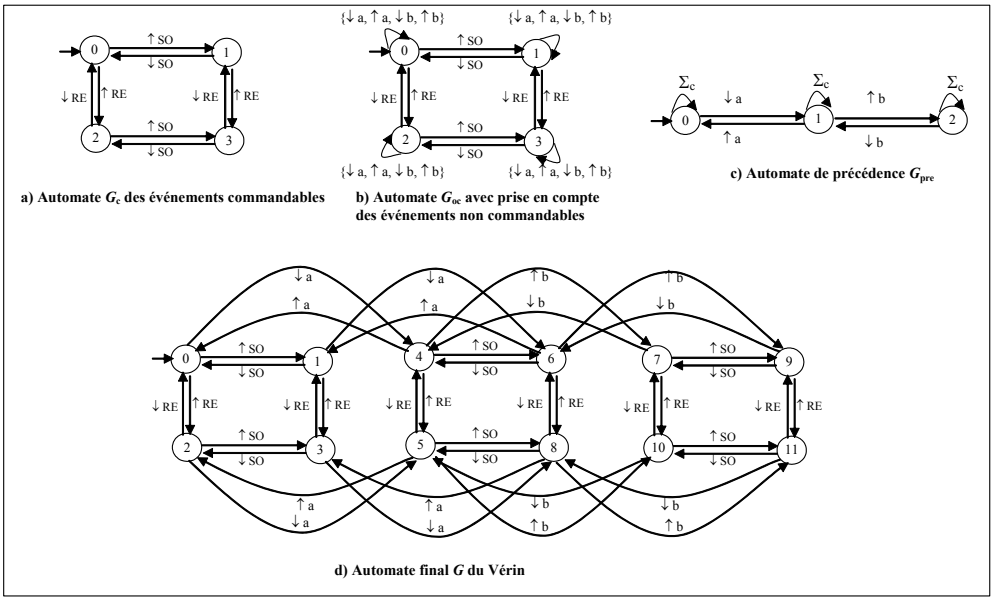

Figure 2-6 : Construction structurée du modèle théorique du vérin

Conditions Initiales	Règles d'occurrence	Relations de précédence
$SO = 0$ $RE = 0$ $b = 0$ $a = 1$	$\uparrow SO \rightarrow \downarrow a$ $\uparrow SO \rightarrow \uparrow b$	$\downarrow a \Rightarrow \uparrow b$
	$\uparrow RE \rightarrow \downarrow b$ $\uparrow RE \rightarrow \uparrow a$	$\downarrow b \Rightarrow \uparrow a$
	$\downarrow SO \rightarrow \downarrow a$ $\downarrow SO \rightarrow \uparrow b$	$\downarrow a \Rightarrow \uparrow b$
	$\downarrow RE \rightarrow \downarrow b$ $\downarrow RE \rightarrow \uparrow a$	$\downarrow b \Rightarrow \uparrow a$

Tableau 2-1 : Interactions et relations entre événements

La comparaison avec le modèle intuitif montre clairement qu'aucune contrainte n'a été prise en compte dans le modèle théorique, car par exemple, pour l'état 3, les deux ordres *SO* et *RE* sont envoyés en même temps à la PO, représentant bien une possibilité de comportement.

Cette modélisation est structurée dans le sens où la construction des modèles se fait de manière logique, appliquée et non plus uniquement intuitive. Cependant, le fait de construire ce modèle automatiquement implique des réactions logiques d'un point de vue théorique qui ne sont plus valables dans la réalité. En effet, chaque EPO dispose de spécifications technologiques qui vont différencier le modèle théorique du modèle pratique. La modélisation pratique constitue alors un compromis idéal entre la modélisation intuitive et théorique. Elle fait appel à la fois à une structure classique et à la connaissance technologique d'un expert.

2.2.3. Modélisation Pratique de la partie opérative

La modélisation pratique d'un EPO se base sur les spécifications technologiques du composant et de son fonctionnement élémentaire. Pour connaître ces caractéristiques, il faut pouvoir étudier la chaîne fonctionnelle d'un procédé. La chaîne fonctionnelle d'un procédé permet d'envoyer des ordres à la PO à partir de la PC par une chaîne d'actions, et de recevoir une information sur la réaction de la PO par une chaîne d'acquisition. La chaîne d'actions est composée de préactionneurs permettant de gérer l'énergie provenant de la PC. Cette énergie est ensuite envoyée aux actionneurs afin de la convertir en action sur l'effecteur. La chaîne d'acquisition permet à travers des détecteurs, voire des transmetteurs, d'acquérir et de transmettre l'information de l'effecteur. Les instructions provenant de la commande sont, par conséquent, envoyées aux préactionneurs qui vont permettre l'alimentation en énergie des actionneurs. Les détecteurs réagissent alors aux différentes actions reçues par les actionneurs. La modélisation d'un Elément de Partie Opérative (EPO) consiste donc à représenter cette chaîne fonctionnelle à travers ces éléments que sont : les préactionneurs, les actionneurs et les détecteurs. Les différents composants élémentaires d'une PO sont obtenus en fonction de leur technologie. Chaque modèle de chaque composant (Figure 2-7) est défini en deux parties :

- Le modèle des détecteurs lié aux entrées de la PC,
- Les modèles préactionneurs et actionneurs liés aux sorties de la PC.

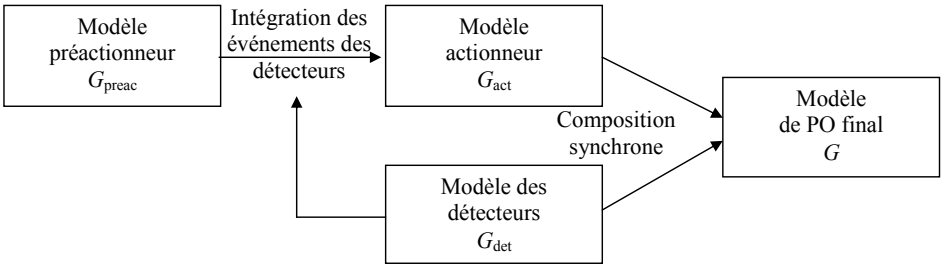

Figure 2-7 : Obtention du modèle pratique de PO

Par exemple, la technologie du distributeur associée au VDE engendre quelques spécificités qui doivent être représentées :

- Une activation puis désactivation d'un même ordre ne change pas la position des chambres du distributeur mais implique une réaction du vérin. Ainsi, une activation de l'ordre *SO* déplace le tiroir du distributeur vers la droite. Si maintenant, on désactive ce même ordre alors le tiroir du distributeur ne bouge pas et entraîne quand même une

sortie du vérin.

- Une priorité sur le premier ordre envoyé existe en cas de conflit de commande. Cela signifie que si l'ordre de sortie *SO* est envoyé avant l'ordre de rentrée *RE*, alors le vérin effectuera une sortie de sa tige.

Le résultat est un automate à 15 états (Figure 2-8) dont le lecteur peut retrouver la construction dans (Philippot, 2006). De l'état 0, l'ordre *SO* peut être activé (état 1) puis désactivé (retour à l'état 0) et entraîne quand même la sortie du vérin. L'activation de l'ordre *SO* (état 1) puis de l'ordre *RE* (état 3) entraîne une sortie du vérin. De la même manière, si l'ordre *RE* est activé de l'état initial puis *SO*, le vérin continue sa rentrée (état 2 puis état 4). Ces deux situations avec les mêmes ordres envoyés ne répondent pas de la même façon et sont donc bien différentes. De même, à partir de l'état 3, si l'ordre *SO*, qui était prioritaire, est désactivé (état 2), alors c'est l'ordre de rentrée *RE* qui devient prioritaire et fait rentrer le vérin puisqu'il est toujours activé.

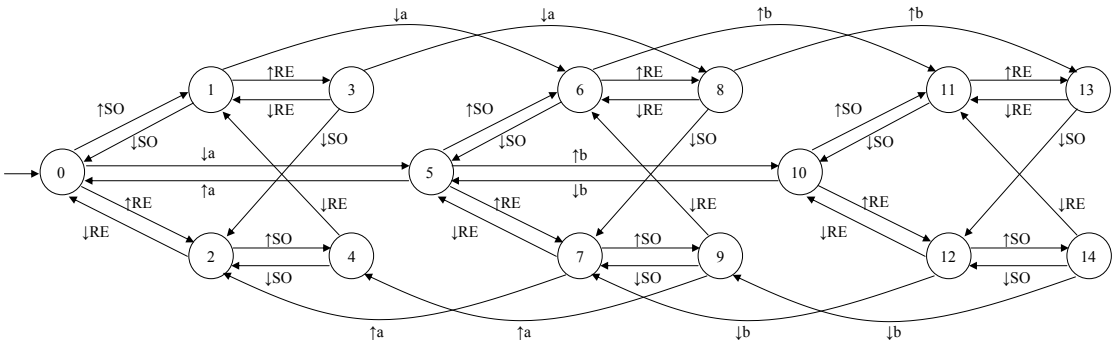

Figure 2-8 : Modèle pratique de la PO du VDE avec distributeur 5/2 bistable

C'est sur cette base de modèles d'EPO qu'a été ensuite développée deux approches de synthèse de commande centralisées permettant de diminuer certaines étapes d'explosion de l'espace d'états.

2.2.4. Approche centralisée par extraction de vivacité

L'approche repose sur une répartition des contraintes de spécifications en contraintes de sécurité (ce que l'on ne doit pas faire) et de vivacité (ce que l'on veut faire). Les contraintes de sécurité locales (à l'EPO) et globales (au système) sont définies pour créer un superviseur global. Ensuite, les contraintes de vivacité locales et globales sont ajoutées pour établir un

contrôleur global qui sera implémenté par la suite dans l'API (Philippot and Tajer, 2010). Cette répartition par extraction de vivacité permet de fractionner la procédure en synthèse locale et globale dans le but de réduire partiellement le problème d'explosion combinatoire (Figure 2-9).

Après les étapes de modélisation du procédé et des contraintes, l'approche consiste à effectuer une synchronisation complète entre tous les EPOs. Sachant que chaque EPO décrit un comportement local possible, la composition des EPOs décrit donc le comportement global possible du procédé, mais rajoute également des interactions impossibles, comme des incohérences. Il en résulte un automate global *G* qui est soumis à une explosion combinatoire d'états en raison de cette composition. Les contraintes de sécurité locales et globales sont appliquées dans un premier temps sur le modèle du procédé global *G* pour obtenir un superviseur global (*SupG*). L'automate résultant de cette première synthèse correspond au comportement maximal commandable par rapport aux contraintes de sécurité. À partir de celui-ci, seul le comportement autorisé par les contraintes de vivacité est maintenu pour obtenir le contrôleur global (*CG*) par extraction de vivacité. Ce *CG* est transformé ensuite en un code respectant la norme (IEC 61131-3) pour l'implémentation dans l'API.

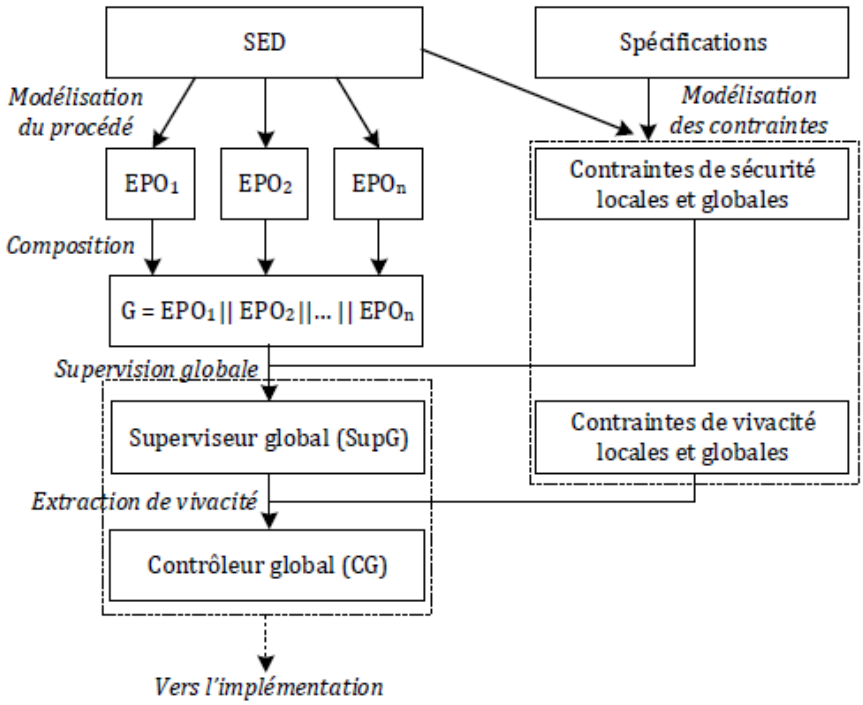

Figure 2-9 : Structure de l'approche centralisée

2.2.5. Approche centralisée raffinée

L'approche précédente permet de réduire en partie le problème d'explosion combinatoire par rapport à l'approche centralisée initiale. Cependant, cette solution oblige à passer par une étape de construction d'un modèle global du procédé par composition des modèles de ces composants. En effet, c'est la constitution d'un modèle global du procédé qui engendre très souvent des problèmes d'explosion combinatoire. Une seconde proposition est de répartir les contraintes locales de sécurité des autres contraintes (Tajer *et al.*, 2011). Cette approche centralisée dite « raffinée » a pour objectif de diminuer l'explosion combinatoire due à l'étape de constitution du procédé global dans sa contrepartie classique. Pour cela, un algorithme applique localement les contraintes de sécurité sur chaque modèle d'EPO avant l'étape de composition (Figure 2-10). Cet algorithme reçoit comme entrées un EPO et l'ensemble de ses contraintes locales de sécurité correspondantes à l'EPO et fournit en sortie un superviseur local (*SupLi*). La composition des superviseurs locaux permet d'obtenir le superviseur global (*SupG*) qui décrit le comportement supervisé global par rapport aux contraintes locales de sécurité. Les contraintes locales de vivacité et les contraintes globales de sécurité et de vivacité sont ensuite appliquées à ce *SupG* pour obtenir le contrôleur global (*CG*) à transformer en programme API d'une manière similaire à l'approche précédente.

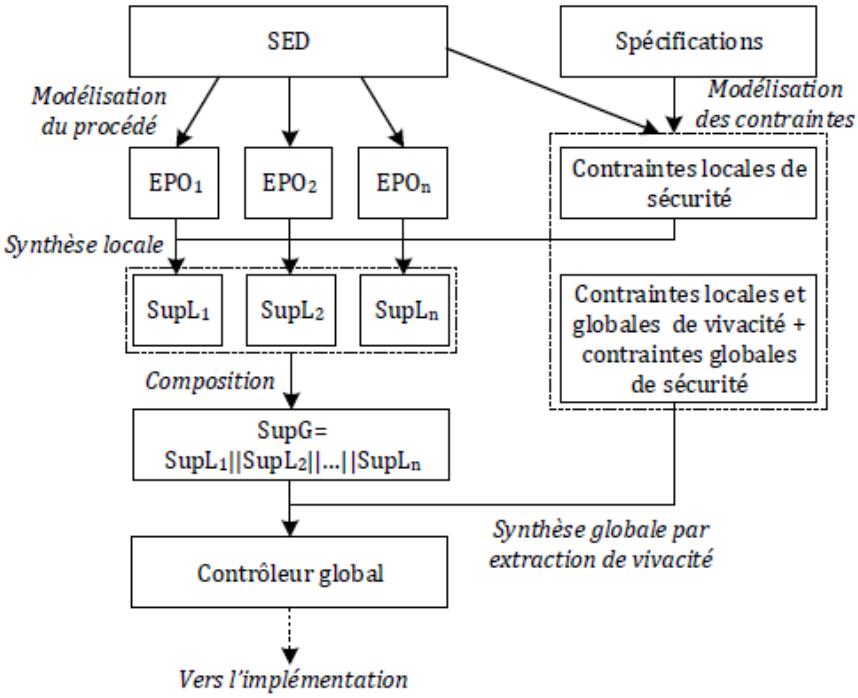

Figure 2-10 : Structure de l'approche centralisée raffinée

Cette proposition raffinée n'évite pas l'explosion combinatoire de l'espace d'états car une étape de composition est toujours présente. Cependant, elle permet de traiter certains systèmes complexes différemment en réduisant en partie cette complexité. Pour contourner ce problème, il semble nécessaire de proposer des approches évitant la composition de modèles telles que les structures décentralisées.

2.2.6. Approche décentralisée

L'approche décentralisée proposée consiste à éviter toutes les étapes de composition qui causent l'explosion combinatoire. Cette approche repose sur les trois premières étapes de l'approche centralisée raffinée : modélisation du procédé, modélisation des contraintes et synthèse des superviseurs locaux. Une extraction de la vivacité locale est réalisée par l'application des contraintes locales de vivacité aux modèles des superviseurs locaux obtenus. Il en résulte des contrôleurs locaux (*CL*) commandant chaque EPO individuellement. L'interaction entre les différents *CL* est prise en compte grâce à un coordinateur central de haut niveau défini à partir des contraintes globales de sécurité et de vivacité et qui joue le rôle d'un filtre autorisant les actions des *CL* (Figure 2-11).

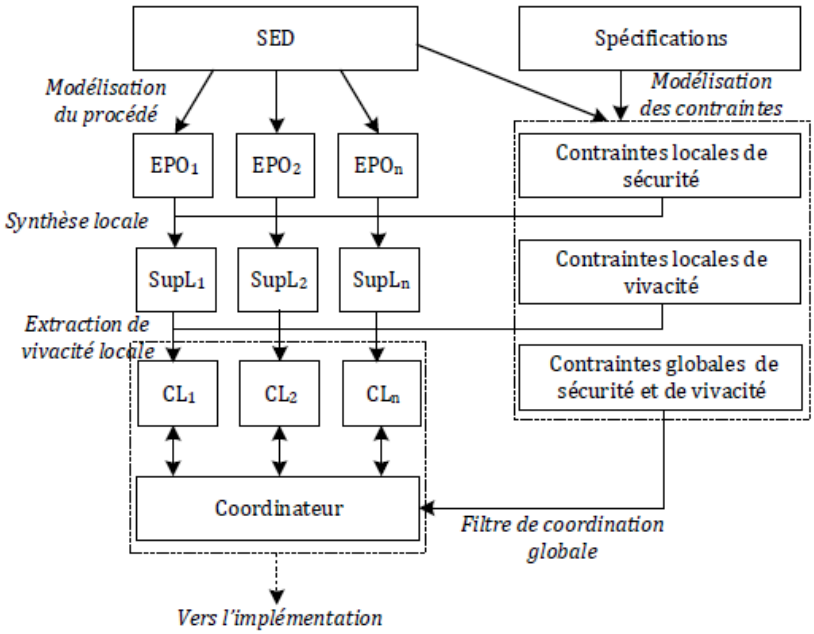

Figure 2-11 : Structure de l'approche décentralisée

Une illustration comparative est disponible pour le lecteur dans (Tajer *et al.*, 2011).

La difficulté de l'approche décentralisée peut alors résider dans l'interaction des modèles de contrôleurs locaux entre eux en passant par le coordinateur. Pour lever cette question, une approche distribuée a été étudiée.

2.2.7. Approche distribuée

La démarche de synthèse de la commande distribuée fait l'objet notamment des travaux de (Qamsane et al*., 2014, 2017(a), 2017(b)). L'architecture de commande proposée est dans la*

Figure 2-12. Elle est divisée en deux parties : (a) la commande supervisée d'un SED selon la théorie de la commande par supervision, et (b) l'approche proposée pour la synthèse et l'implémentation hors ligne de la commande distribuée. Elle est basée sur quatre étapes principales :

- Une synthèse locale de la commande : Obtention d'un Contrôleur Local (CL) pour chacun des composants (EPO).
- Une synthèse globale de la commande : Synchronisation interactive entre les EPO en vue de réaliser un objectif global de commande. Elle permet donc de prendre en compte les interactions entre les composants distribués de la PO. Cette étape consiste à appliquer les contraintes globales aux CL obtenus par synthèse locale. Le modèle de base de la commande obtenue est appelé automate de Contrôleur Distribué (CD).
- Une vérification du non-blocage et de la vivacité : Levée des conflits dus à l'utilisation de 2 formalismes (automates avec des expressions booléennes) pour effectuer une synthèse distribuée. Dans cette troisième étape, la technique de *model-checking* est utilisée pour vérifier l'absence de blocages dans la commande distribuée. Parmi les différentes techniques de vérification formelle, le *model-checking* (Baier and Katoen, 2008) semble efficace dans la vérification des conceptions développées. Il consiste à calculer partiellement l'ensemble des transitions d'états accessibles d'un modèle, puis d'évaluer si une contrainte formelle exprimée en logique temporelle est valable pour cet ensemble.
- Une implémentation en GRAFCET : Interprétation des CD en modèles GRAFCET (IEC 60848) à des fins d'implémentation dans un API.

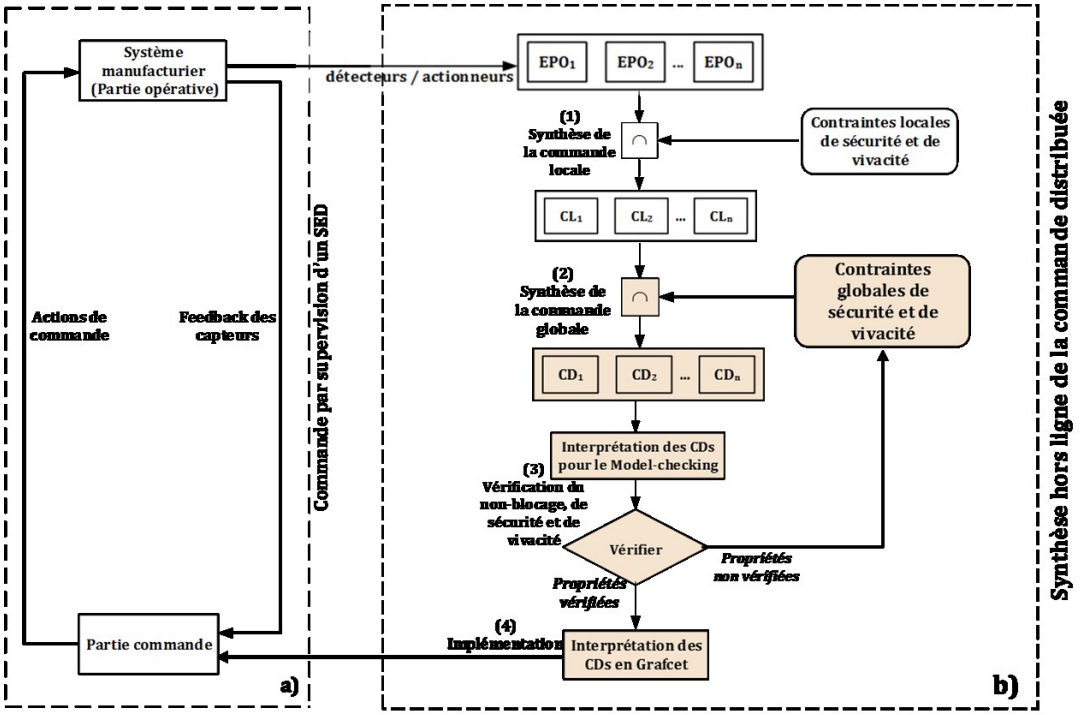

Figure 2-12 : Architecture pour la synthèse de la commande distribuée

Cette approche a fait l'objet de 2 revues internationales (Qamsane *et al.*, 2017(a), 2017(b)). J'invite également le lecteur à suivre la thèse de Monsieur Y. Qamsane afin de disposer plus des détails scientifiques de l'approche.

L'applicabilité et l'efficacité de l'approche pour des systèmes industriels ont été démontrées par l'application sur un système manufacturier réel de taille importante. Une étude comparative entre l'approche distribuée et l'approche centralisée a été réalisée afin d'évaluer la taille du superviseur et l'utilisation de la mémoire de l'API. Les résultats montrent que la commande obtenue par l'approche distribuée a le même comportement que la commande monolithique.

L'application sur le système manufacturier réel nous a permis également d'établir un constat : lorsque le fonctionnel du système est modifié, il convient de reprendre l'approche à partir de l'application des contraintes de vivacité. Bien que la flexibilité soit garantie, il apparait difficile de proposer des solutions de recomposition globale à des industriels. Il convient peut-être de repenser les approches initiales en travaillant davantage sur cette manipulation entre le fonctionnel et la sécurité, par exemple sur l'ensemble des contraintes.

Attention, ces contraintes utilisent souvent des manipulations formelles qui ne sont pas faciles à utiliser et à comprendre puisqu'elles ne sont pas familières aux utilisateurs. Il convient alors de les accompagner dans la méthodologie de conception.

2.3. Approches par contraintes logiques

Pour répondre aux enjeux de sûreté de fonctionnement dans le domaine des systèmes de contrôle-commande, la norme (IEC 61508) et ses déclinaisons sectorielles recommandent la mise en place de méthodes formelles permettant de garantir un processus sûr de développement. Les réponses scientifiques suggèrent habituellement deux approches (Faure and Lesage, 2001) : i) synthétiser la commande en utilisant la théorie de supervision de Ramadge et Wonham (Ramadge and Wonham, 1989), ii) vérifier formellement le modèle de contrôle-commande par *model-checking* (Bérard *et al.*, 1999) ou par calcul symbolique (Roussel and Denis, 2002). Les inconvénients de ces approches se situent à plusieurs niveaux :

- le contrôle-commande est vérifié à partir de modèles de la commande et non celle réellement implantée, l'implémentation d'un modèle passant par une interprétation du programmeur,
- le contrôle-commande est conçu pour une utilisation donnée. Si un changement est effectué dans le programme de l'API lors de la production par l'équipe de maintenance ou que l'opérateur a un rôle décisionnel dans le pilotage du système, la vérification du contrôle-commande faite avant l'implémentation n'a plus d'intérêt par rapport au contrôle-commande utilisé réellement.

A ces incertitudes sur la commande réellement implantée, s'ajoutent des incertitudes liées au pilotage par un opérateur humain. Pour résoudre ce problème, le développement d'approches de surveillance (Chaillet-Subias, 1995, Combacau, 1991) inhibant certaines requêtes de l'opérateur, définies comme non conformes aussi bien au niveau de la sécurité qu'au niveau fonctionnel est proposé. Allant dans ce sens, plusieurs approches de surveillance du comportement de la partie opérative ou de la partie commande ont été développées, par exemple par comparaison avec un modèle de référence du comportement normal du système (Chaillet-Subias, 1995, Combacau, 1991), ou bien par comparaison avec une émulation du comportement de la PO (Holloway and Krogh, 1990, Roussel and Denis, 2002), ou enfin par

filtre interposé entre la PO et la commande (Alanche *et al.*, 1986, El-Khattabi, 1993, Lhoste, 1991, Marangé *et al.*, 2007).

L'utilisation de telles approches permet donc une protection immédiate des équipements, des personnes, des biens et de l'environnement vis-à-vis d'actions incohérentes – voire dangereuses – de la part d'un opérateur ou de la partie commande, en détectant et en empêchant les mauvaises manipulations qui pourraient soit détériorer le système, soit contredire le séquencement souhaité par le cahier des charges. C'est à partir de ce constat que plusieurs travaux ont été menés au sein du laboratoire. D'abord au travers de la thèse de P. Marangé (Marangé, 2008) qui a proposé un filtre de sortie de commande afin de bloquer le système avant une anomalie de commande. Puis, dans les travaux de thèse de R. Coupat (Coupat, 2014) où il modifie le filtre afin de résoudre de façon déterministe un ensemble de contraintes de sécurité. Enfin, R. Pichard (Pichard *et al.*, 2018) a formalisé les contraintes logiques et la notion de priorité, il a défini une propriété de cohérence et de suffisance du filtre, et a proposé des algorithmes de recherche de solution à base de solveur SAT. Dans la suite de cette section, je présente succinctement le concept de filtre bloquant de (Marangé, 2008) avant de reprendre les travaux des 2 doctorants encadrés.

2.3.1. Filtre bloquant

La méthode proposée par (Marangé, 2008), pour sécuriser les ordres envoyés de la PC vers la PO, est basée sur l'utilisation d'un ensemble de contraintes de sécurité qui agit comme un filtre logique s'exécutant à la fin du programme de l'automate programmable industriel et qui interdit les états dangereux (Figure 2-13).

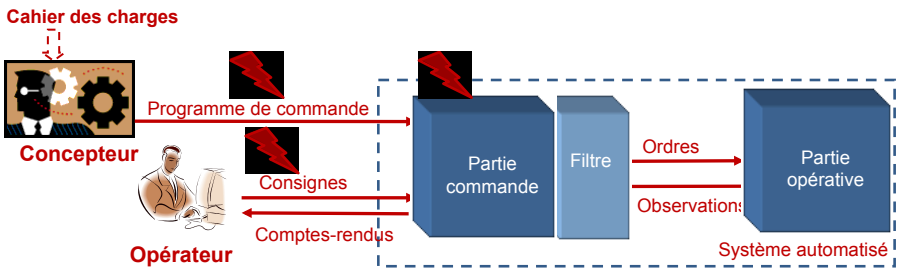

Figure 2-13 : Principe du filtre de sécurité (Marangé, 2008)

Le filtre est un ensemble de contraintes de sécurité (CS) obtenu à partir d'une approche

modulaire et itérative de modélisation du procédé et de sa validation. Cela permet de diminuer la complexité, et d'assurer son applicabilité dans le monde de l'industrie. Le filtre est conçu par un expert et un ingénieur conçoit la commande de façon classique. Le filtre stoppe le système dans un état sûr dès qu'une contrainte n'est pas respectée. L'expert mène une analyse dysfonctionnelle (au moyen d'une AMDEC par exemple) pour définir l'ensemble des états dangereux à ne pas atteindre pour les éléments de PO et les produits.

À chaque état dangereux va correspondre un ensemble de contraintes à respecter. L'expert identifie pour chaque état dangereux le sous-ensemble de la PO (les composants et la séquence du flux des produits) à modéliser pour la vérification par *model-checking*. L'expert définit pour chaque sous-système l'ensemble des contraintes sécuritaires qui constitue le filtre et identifie aussi les observateurs nécessaires à la reconstruction binaire des informations manquantes (non-observabilité d'une pièce entre deux capteurs par exemple).

Initialement, le filtre avait été prévu pour stopper la commande en bloquant les actionneurs dans un état de sécurité et empêcher ainsi l'évolution de la PO. Cette approche a été étendue pour faire de la commande sûre de fonctionnement (Benlorhfar *et al.*, 2011). L'idée consiste en cas de violation d'une contrainte de sécurité à corriger le ou les ordres erronés de commande uniquement pour le cycle API en cours. Ainsi, à la façon de la SCT, le filtre autorise ou inhibe les événements commandables. Cette approche présente des avantages. Il devient possible de séparer structurellement les parties « sécuritaire » et « fonctionnelle » dans le contrôleur. Cela peut éventuellement permettre une simplification de la loi de commande mais change radicalement la façon de travailler des automaticiens. Toutefois, cette méthode peut également être utilisée pour sécuriser des programmes existants (Riera *et al.*, 2015b).

2.3.2. Génération d'un filtre correcteur

L'approche par filtre bloquant permet la vérification formelle hors-ligne de programme automate existant, ainsi que la détection en-ligne d'erreurs provenant de la partie commande. Néanmoins, le blocage systématique du système lors de la détection d'une erreur peut entrainer des arrêts de production trop fréquents. Afin de diminuer le risque de blocage du système, (Benlorhfar *et al.*, 2011) ont proposé la notion de filtre correcteur.

Le principe est de modifier l'implémentation du filtre afin de proposer en-ligne des valeurs de sortie alternative, basées sur les valeurs envoyées par la partie commande, qui ne violent aucune contrainte de sécurité. Dans ces conditions, le système n'est pas forcément bloqué et peut potentiellement continuer la production dans un mode dégradé. Avec le filtre correcteur, il est donc possible de modifier dynamiquement la commande sans pour autant forcément bloquer le système. Néanmoins, avec l'implémentation séquentielle proposée dans (Benlorhfar *et al.*, 2011), la résolution d'une contrainte peut violer une contrainte précédemment non-violée ou résolue. De plus, en fonction de l'ordre dans lequel les contraintes sont implémentées, la solution n'est pas obligatoirement la même, la résolution n'est donc pas déterministe. Afin de résoudre ces problèmes, (Coupat, 2014) a proposé un algorithme itératif permettant la résolution déterministe d'un ensemble de contraintes de sécurité. Enfin, une redéfinition des contraintes a été proposée dans le but d'étendre l'approche par filtre à la commande sûre de fonctionnement. Les deux nouveaux types de contraintes, remplaçant ceux de Marangé, sont :

- les contraintes simples : ne contiennent qu'une seule variable de sortie.
- les contraintes combinées : contiennent plusieurs variables de sorties.

Par ailleurs, Coupat propose d'utiliser ces contraintes pour deux notions différentes : les contraintes de sécurité et les contraintes fonctionnelles. Les premières ont pour objectif de formaliser les exigences de sécurité du cahier des charges (configurations interdites du système), les secondes ont pour objectif de formaliser la partie fonctionnelle (configurations souhaitées du système).

Les contraintes sont toujours définies du point de vue de la partie commande (PC), et il est supposé que le temps de cycle API est suffisant pour détecter tous les changements du vecteur d'entrées (opération synchrone, changements d'états simultanés des entrées de l'API possibles). Dans cette approche, les contraintes de sécurité sont exprimées sous la forme de monômes logiques (produit des variables logiques, $\prod$). Les contraintes de sécurité doivent toujours être égales à 0 (*FAUX*) à la fin de chaque cycle API avant la mise à jour des sorties (Figure 2-14).

Les contraintes doivent être également définies afin de garantir la contrôlabilité du système. Cela signifie que l'espace sécuritaire du système doit être suffisamment important

pour que le système soit vivant. Après application des contraintes de sécurité, il doit être possible de concevoir un contrôleur qui répond aux spécifications fonctionnelles du système. Par exemple, en considérant l'hypothèse précédente au sujet de l'état initial sûr, un ensemble de contraintes de sécurité, qui remet à zéro toutes les sorties, est sûr mais n'assure pas la contrôlabilité.

Il faut différencier les contraintes qui impliquent une sortie unique à l'instant *t* (appelées contraintes de sécurité simples, *CSs*), et les contraintes impliquant plusieurs sorties à l'instant *t* (appelées contraintes de sécurité combinées *CSc*). Les contraintes exigent la connaissance des E/S à l'instant *t* et aux instants précédents (présence de fronts par exemple).

Il peut être nécessaire de définir des observateurs en raison du manque d'observabilité du système. A titre d'exemple, une caisse peut être présente sur un convoyeur sans qu'un capteur indique sa présence. Les observateurs correspondent idéalement à une fonction séquentielle des entrées de l'API et permettent d'intégrer un état de la séquence aux contraintes de sécurité. L'ensemble de contraintes de sécurité est considéré comme nécessaire et suffisant pour garantir la sécurité du système. En d'autres termes, le retrait d'une contrainte ne garantit plus la sécurité et l'ajout d'une contrainte n'apporte rien au niveau de la sécurité. Dans cette approche, les contraintes de sécurité peuvent toujours être représentées comme des monômes logiques et dépendent des entrées (à *t*, *t*-1, *t*-2…), sorties (à *t*, *t*-1, *t*-2…) et observateurs (dépendant uniquement des entrées à *t*, *t*-1, *t*-2…).

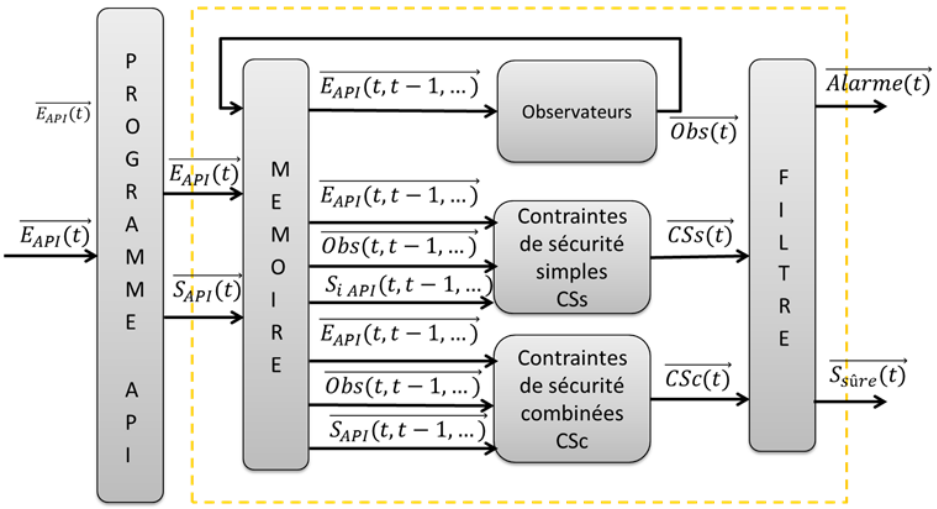

Figure 2-14 : Structure interne du filtre logique (Coupat, 2014)

L'algorithme de commande proposé par R. Coupat sépare les exigences de sécurité des exigences fonctionnelles. Ces dernières vont être formalisées au moyen de contraintes fonctionnelles (CF). Toute contrainte qui ne fait pas partie de l'ensemble des contraintes de sécurité est considérée comme une CF. L'algorithme de contrôle proposé consiste à chaque cycle API, à autoriser les commandes fonctionnelles compatibles avec les exigences de sécurité. En d'autres termes, il est possible de considérer une commande existante comme générant les contraintes fonctionnelles. Si la commande est bien faite, en théorie aucune contrainte de sécurité ne sera violée. Dans le cas contraire, la sécurité restera tout de même assurée sans toucher à la commande existante. Cela signifie que l'algorithme de commande sûre développé peut être intégré sans difficulté dans un *template de commande à base de contraintes*. L'algorithme ne sera pas développé dans cette section. Le lecteur peut se référer à la thèse de R. Pichard qui a proposé une évolution et une formalisation du filtre logique (Pichard, 2018).

Les travaux de Coupat ont tout de même permis une première application industrielle de l'approche par filtre de commande avec la SNCF.

Par ailleurs, dans les travaux de Marangé une propriété de suffisance de l'ensemble des contraintes a été proposée. Dans l'algorithme de Coupat, s'il n'existe pas de solution lors de la résolution, alors une solution arbitraire va être appliquée, mais cette solution ne vérifie pas l'ensemble des contraintes. L'expression d'une propriété d'existence de solution est donc nécessaire.

La notion de cohérence d'un ensemble de contraintes de sécurité a été proposée dans (Riera *et al.*, 2015a). Un ensemble de contraintes de sécurité est cohérent si et seulement si, quelle que soit la valeur des entrées du filtre (capteurs, variables internes, commandes envoyées par le programme amont), il existe toujours une solution (valeurs des commandes) vérifiant l'ensemble des contraintes. Un ensemble de conditions nécessaires à la cohérence d'un ensemble de contraintes de sécurité a été également étudié.

2.3.3. Méthodologie de conception du filtre logique

L'objectif de l'approche proposée dans (Pichard *et al.*, 2018) est de fournir une méthode et des outils utilisables dans l'industrie permettant, à partir du cahier des charges, d'obtenir un

programme API généré automatiquement et vérifié formellement (Figure 2-15).

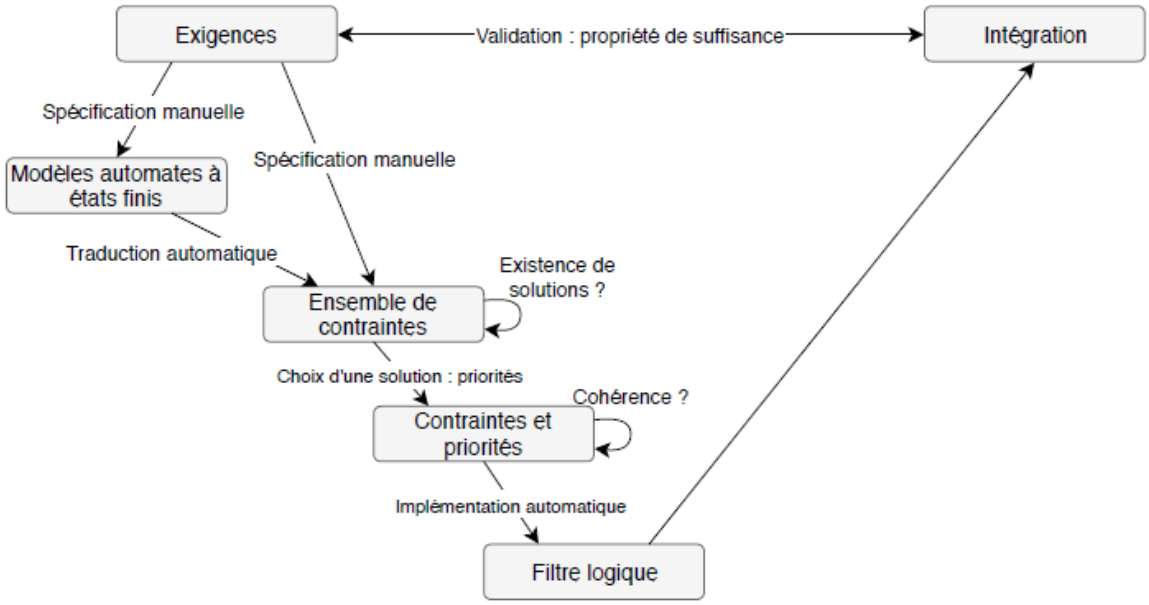

Figure 2-15 : Approche formelle proposée pour la conception d'un filtre (Thèse R. Pichard)

Une des premières contributions de cette thèse est de pouvoir vérifier l'existence de solutions à partir d'un ensemble défini de contraintes (vecteurs de sortie validant toutes les contraintes), et ce quelle que soit la valeur des variables d'entrées du filtre. Une fois cette première étape réalisée, et dans le but d'obtenir une solution déterministe, le choix d'une solution particulière doit être effectué. Ce choix revient à imposer des priorités entre les variables de sorties (actionneurs) de certaines contraintes logiques. Il convient ensuite de vérifier formellement que les contraintes avec les priorités choisies sont cohérentes entre elles, mais également nécessaires et suffisantes. R. Pichard a ensuite développé un outil logiciel (SEDMA) de génération automatique du code filtre à implémenter dans un API. Afin d'appliquer les différentes étapes de l'approche, un effort important de formalisation a été apporté (Pichard, 2018).

L'approche proposée permet l'obtention d'une condition nécessaire et suffisante à la cohérence d'un ensemble de contraintes logiques (Figure 2- 16). Dans cette approche de vérification, un problème correspond à un ensemble de contraintes logiques avec les priorités fonctionnelles associées.

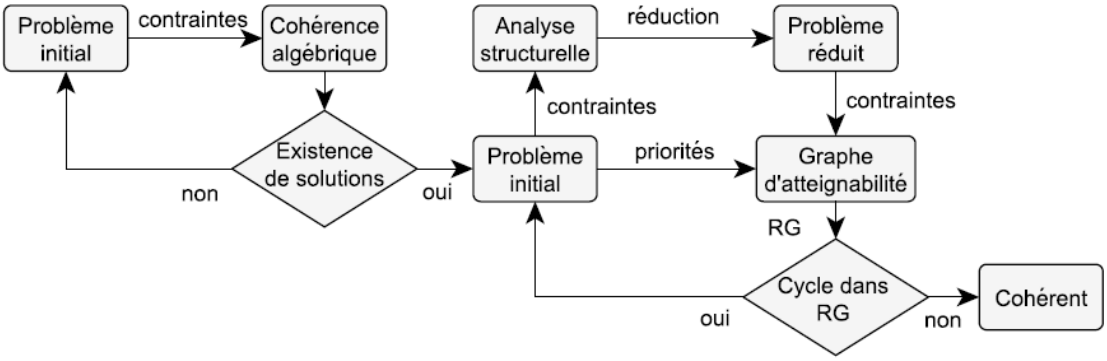

Figure 2- 16 : Méthode de vérification formelle de la cohérence

L'approche débute par une vérification de la cohérence algébrique en considérant les contraintes sans les priorités fonctionnelles. Celle-ci est effectuée à l'aide de la synthèse algébrique (Hietter, 2009). Elle permet de vérifier l'existence d'un espace de solution. Si au moins une solution existe, alors l'analyse peut continuer sous deux étapes. Premièrement, une analyse structurelle des contraintes est effectuée, permettant de réduire le nombre de contraintes à vérifier. Basé sur ce problème réduit, un graphe d'atteignabilité (RG) est construit afin de mettre en évidence des cycles d'évolutions. Ces cycles représentent une condition nécessaire et suffisante à la cohérence d'un ensemble de contraintes logiques quelconques.

Exemple : Soit l'ensemble de contraintes suivant avec {a, b} des événements non-commandables et {O1, O2, O3} des événements commandables :

- $CSs1 = a \cdot O2$
- $CSs2 = \neg a \cdot b \cdot O1$
- $CSc1 = O1 \cdot \neg O2$
- $CSc2 = O2 \cdot \neg O3$

Le graphe structurel de la Figure 2-17 permet de montrer que la contrainte CSs2 n'est pas en interaction avec les autres et permet de sortir de l'analyse d'atteignabilité. En effet, une arête existe entre deux contraintes uniquement si : i) leurs parties non-commandables respectives peuvent être vraies simultanément, ii) il existe au moins une variable commandable sous forme complémentée dans l'une des contraintes et non complémentée dans l'autre. Dans l'exemple, c'est l'événement O2 qui provoque le lien entre CSs1, CSc1 et CSc2.

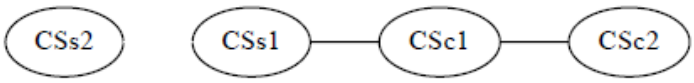

Figure 2-17 : Graphe structurel de l'exemple

Ce graphe doit contenir l'ensemble des chemins de résolution possibles, c'est-à-dire comment, à partir d'une affectation non-valide, il est possible d'atteindre un état valide (Figure 2-18). Le calcul des affectations valides est notamment possible avec un solveur SAT (Du *et al.*, 1997). Si pour toutes les affectations non-valides il est possible d'atteindre un état valide, alors le problème est cohérent par définition. A contrario, le problème est non-cohérent si et seulement si il existe au moins un cycle dans le graphe d'atteignabilité.

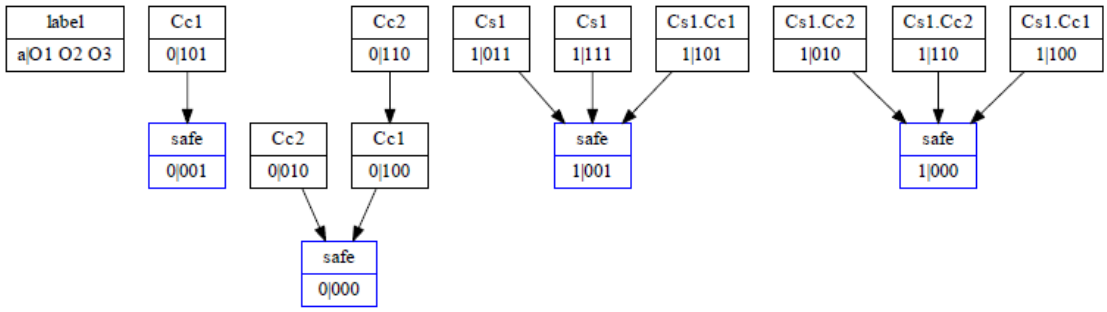

Figure 2-18 : Graphe d'atteignabilité de l'exemple

R. Pichard a ensuite proposé et comparé 3 algorithmes d'implémentation du filtre logique. Le but est de proposer des algorithmes, implémentables dans un automate programmable industriel, permettant de trouver en-ligne une affectation des sorties respectant toutes les contraintes. De plus, ces algorithmes doivent être déterministes, c'est-à-dire qu'à un vecteur d'entrée donné, le vecteur de sortie doit toujours être le même. Les caractéristiques de ces algorithmes sont :

- Algorithme itératif : algorithme issu de R. Coupat, peu complexe, mais avec une hypothèse de cohérence des contraintes avec priorités ;
- Algorithme de Hamming : sous hypothèse de cohérence algébrique, technique de recherche locale basée sur la distance de Hamming ;
- Solveur SAT : sous hypothèse de cohérence algébrique, technique à base de solveur SAT (propagation de contraintes, *back-tracking*), recherche locale, heuristique de choix (Hamming).

Le lecteur retrouvera un ensemble de critères comparatifs dans le chapitre IV de la thèse de R. Pichard (Priorités, cohérence, implémentation, maintenabilité, espace mémoire, adaptabilité, complexité). Ce travail a notamment été validé sur 3 stations de la cellule CellFlex. Une expérimentation a également été menée auprès d'étudiants et de collègues afin de montrer les (non-)pratiques en méthodologie de conception. Cette partie, en lien direct avec les difficultés de l'opérateur humain devant des méthodologies scientifiques, sera rediscutée dans le chapitre suivant.

3. Diagnostic des SAP

Maintenabilité et disponibilité des équipements sont les deux mots d'ordre de l'industrie depuis quelques années. Il existe une nécessité de connaître l'état de ses installations afin de prédire et/ou prévenir au plus tôt un comportement défaillant. Les politiques de maintenance évoluent et l'on parle aujourd'hui de diagnostic et pronostic de systèmes. Une réelle problématique d'observation du comportement de systèmes complexes est présente. La plupart des travaux de la littérature autour de cette problématique se base sur l'utilisation de modèles observateurs appelés « diagnostiqueurs » (Sampath, 1995). Cependant, l'obtention de tels modèles nécessite une reconstruction du comportement normal mais aussi anormal du système sous observation totale ou partiale du système (Lefebvre, 20014). Il convient alors de ne pas surexposer ces modèles à l'explosion combinatoire inhérente aux Systèmes à Evénements Discrets.

Cet axe de recherche s'inscrit dans la thématique générale de la sûreté de fonctionnement des systèmes et est en interaction avec l'axe sur la synthèse de la commande (Fri *et al.*, 2016). Il s'appuie sur mes travaux de thèse mais également sur les collaborations que j'ai pu ensuite entreprendre en interne ou au niveau national. En base bibliographique, j'invite le lecteur à regarder le tour d'horizon des approches de diagnostic proposé dans (Zaytoon and Lafortune, 2013).

3.1. Résumé des travaux initiaux

Le diagnostic des défaillances est indispensable à la bonne conduite d'un Système Automatisé de Production. Les systèmes manufacturiers sont constitués de différents éléments technologiques devant réagir entre eux et constituant ce qu'on appelle la Partie Opérative PO.

On parle alors de systèmes informationnellement décentralisés sur chaque élément.

La problématique du diagnostic des SED est essentiellement liée au fait que les modèles représentant les SED ne sont pas riches en terme d'informations (Figure 2-19). Par conséquent, il faut exploiter toute l'information disponible sur le procédé. Il s'agit de prendre en compte à la fois :

- Le comportement événementiel de la PO à travers toutes ses situations possibles. Son modèle suppose une connaissance précise de la technologie du matériel du point de vue de son architecture.
- Les spécifications obtenues à partir d'un cahier des charges. L'ensemble de ces spécifications est écrit par un langage et traduit sous forme d'un modèle de commande utilisé par la PC.
- L'information temporelle qui représente la réactivité des actionneurs à un ordre envoyé par la PC. Elle peut être représentée par des contraintes temporelles, comme par exemple les chroniques ou les *templates*.
- Le modèle symbolique qui décrit les réactions particulières et les contraintes globales du procédé qui ne sont incluses dans aucun des modèles précédents. Il doit analyser les situations ou comportements incohérents à partir d'une expertise, de documentation, du cahier des charges, d'historiques…

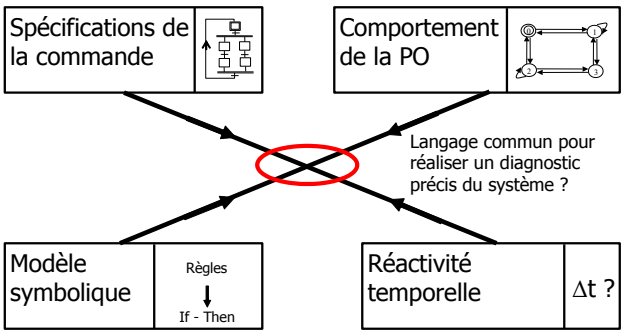

Figure 2-19 : Problématique du diagnostic des SED

Pour répondre à cette problématique, une démarche de construction d'un diagnostic décentralisé pour les Systèmes à Evénements Discrets a été présentée durant mon doctorat. Cette démarche globale s'effectue pour une part, hors ligne à travers la construction de diagnostiqueurs locaux et du coordinateur en 6 étapes (Figure 2-20) et, pour une autre part, en ligne pour son implantation.

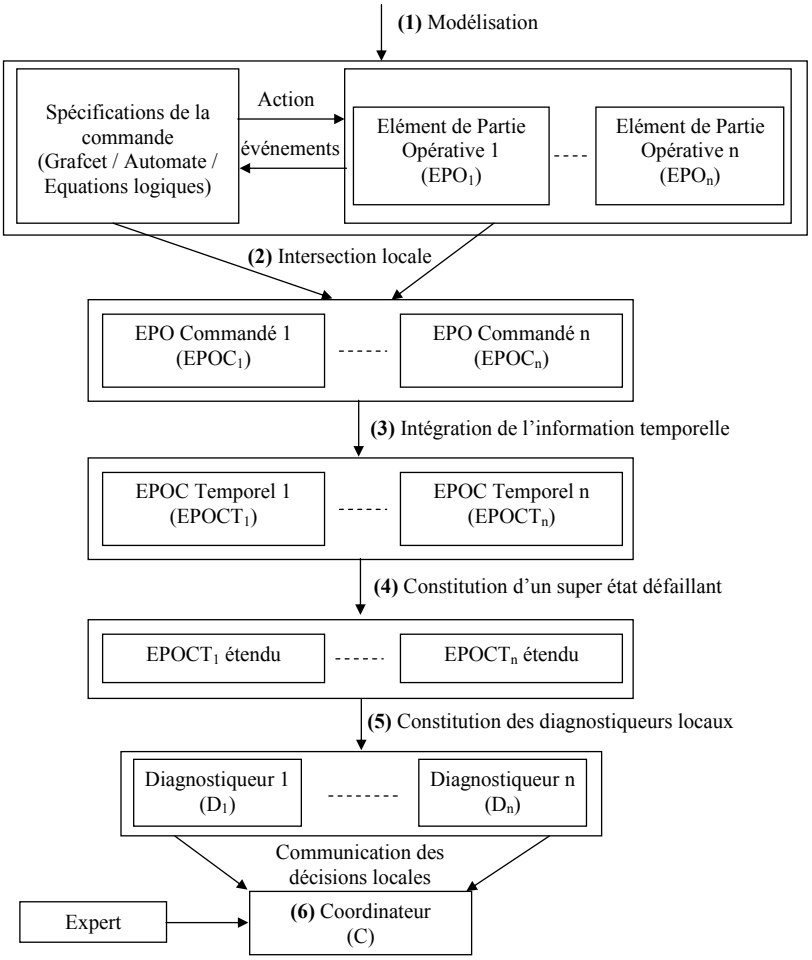

Figure 2-20 : Démarche de diagnostic décentralisé avec coordinateur (Philippot, 2006)

1) Modélisation

L'étape de modélisation peut être composée en deux phases :

- L'élaboration des spécifications de la commande afin de représenter le comportement désiré du procédé. Ces spécifications sont représentées soit par une commande en GRAFCET, soit par des contraintes de vivacité et de sécurité exprimées sous forme d'automates ou d'expressions logiques.
- La modélisation de la partie opérative sous forme modulaire pour représenter le comportement logique de chaque élément par un modèle à base d'automate à états, appelé un Elément de Partie Opérative.

La modélisation de la PO s'effectue sous forme d'automates décrivant les évolutions physiquement possibles du procédé par des événements élémentaires. L'ensemble des événements Σ peut être décomposé selon la notion d'observabilité. Ainsi, les événements liés

aux capteurs et actionneurs d'un procédé pouvant être observés appartiendront à l'ensemble des événements observables $\Sigma_o \subseteq \Sigma$ alors que les événements de défauts ou n'appartenant pas à l'élément de PO seront considérés comme non observables $\Sigma_{uo} \subseteq \Sigma$. Cette notion d'observabilité est donc essentielle au diagnostic des défauts.

Une autre caractéristique importante pour un procédé est de pouvoir également distinguer les événements qui sont commandables de ceux qui ne le sont pas. En effet, les événements ne sont pas toujours générés de façon spontanée, mais réagissent à des commandes d'entrée, générant des réponses en sortie. Les interactions entre la PC spécifiée et la PO sont alors mises en évidence d'un point de vue commande. Par conséquent, les ordres sont des événements envoyés par la PC et permettent de commander le procédé. Ils sont vus comme des sorties de la PC, ou comme les entrées de la PO, et sont des événements commandables $\Sigma_c \subseteq \Sigma_o$. A contrario, les événements provenant des capteurs répondent à une action en entrée de la PC, ou comme les sorties de la PO, et sont donc non commandables $\Sigma_{uc} \subseteq \Sigma_o$ (Balemi *et al.*, 1993).

Par ailleurs, un événement commandable correspond soit à l'activation, $\uparrow z$, soit à la désactivation, $\downarrow z$, d'un ordre « z » de la commande, tandis qu'un événement non commandable est associé soit au front montant, $\uparrow e$, soit au front descendant, $\downarrow e$, d'une variable d'entrée « e » de la PC. Un front montant correspond au passage de 0 à 1 d'un signal binaire, alors qu'un front descendant est considéré comme le passage de 1 à 0 de ce signal binaire. Les ensembles Σ_c et Σ_{uc} s'écrivent alors $\Sigma c = \uparrow Z \cup \downarrow Z$ et $\Sigma_{uc} = \uparrow E \cup \downarrow E$, où $\uparrow Z$ et $\downarrow Z$ correspondent aux activations et désactivations de l'ensemble de tous les ordres de la PC et $\uparrow E$ et $\downarrow E$ reflètent, respectivement, les fronts montant et descendant de l'ensemble de tous les événements non commandables.

2) Intersection locale

Cette étape consiste à intégrer les informations du cahier des charges aux différents EPO. Elle se compose de deux étapes :

- La transformation de la spécification de la commande exprimée en GRAFCET en un Graphe Equivalent (GE) permettant d'avoir une sémantique identique à celle des modèles d'EPO.

- L'intersection locale entre le GE et chacun des EPO afin d'obtenir le comportement désiré local : modèles EPO Commandés EPOCi ($i \in \{1, 2, \ldots, n\}$).

3) Intégration de l'information temporelle

L'étape d'intégration de l'information temporelle consiste à prendre en compte les contraintes temporelles qui s'exercent sur les EPOC. L'évaluation de ces contraintes est réalisée par des fonctions de prévision basées sur la logique floue. Cela signifie que chacune de ces fonctions peut prendre toutes les valeurs entre 0 et 1 et non pas simplement 1. Elle indique soit la violation d'une contrainte temporelle (valeur 1) soit le respect de la contrainte (valeur 0). L'intérêt de l'utilisation de la logique floue est de prévoir l'évolution vers un défaut par l'augmentation de la valeur fournie par ces fonctions de prévision de 0 vers 1. La prise en compte de cette information temporelle aux différents EPOC conduit vers des modèles d'EPOC Temporels : EPOCTi ($i \in \{1, 2, \ldots, n\}$).

4) Constitution d'un super état défaillant

Cette étape consiste à créer un super état de défaut XF aux EPOCTi, $i \in \{1, 2, \ldots, n\}$. Ce super état de défaut est atteint soit par l'occurrence d'un événement non attendu, soit par la violation d'une fonction de prévision. Son rôle est de détecter les défauts susceptibles de survenir sans pour autant en diagnostiquer la cause. Il en résulte un modèle d'EPOCTi dit modèle EPOCT étendu, avec $i \in \{1, 2, \ldots, n\}$, pour chaque élément i.

5) Constitution des diagnostiqueurs locaux

Pour chaque EPOCTi étendu, un diagnostiqueur local Di doit être créé. Ce diagnostiqueur doit permettre de localiser chaque défaut détecté et de l'identifier. L'ensemble des partitions de défauts est alors établi pour chaque diagnostiqueur local. A partir de chaque état de l'EPOCTi étendu, une fonction de décision comprenant une étiquette et indiquant la situation fonctionnelle du procédé est ajoutée.

6) Construction du coordinateur

Cette étape consiste à créer un coordinateur qui va gérer les cas d'indécision entre les diagnostiqueurs locaux, liés à l'observation partielle du procédé. Ce coordinateur est basé sur un ensemble de règles de fusion des décisions locales et un ensemble de règles permettant le diagnostic des défauts qui ne peuvent pas être diagnostiqués par au moins un des

diagnostiqueurs locaux parce qu'ils nécessitent la prise en compte des interactions entre ceux-ci. Ce coordinateur doit assurer un diagnostic équivalent à celui d'un diagnostiqueur global.

3.2. Interaction Filtre-Diagnostic

L'utilisation d'approches de commandes sûres de fonctionnement comme vu plus haut permettent une protection immédiate des équipements, des personnes, des biens et de l'environnement vis-à-vis d'actions incohérentes – voire dangereuses – de la part d'un opérateur ou de la partie commande. En cas d'actions incohérentes, il est alors nécessaire de fournir à l'opérateur une explication compréhensible lui permettant de comprendre son erreur et d'améliorer son expertise pour utiliser le système. Cependant, des pannes sur les capteurs ou les actionneurs de la PO peuvent également survenir, et il apparait comme évident qu'un système de diagnostic pourrait d'une part être couplé au filtre de sécurité pour que ce dernier puisse fonctionner correctement et, d'autre part, coopérer avec un opérateur de conduite qui pourrait statuer sur des indécisions issues du diagnostiqueur. C'est dans ce cadre qu'a été conduite une collaboration avec le CRAN de Nancy et le LAMIH de Valenciennes autour du projet GIS ADEXEC (Approche de Détection et d'EXplication d'Erreur de Commande par filtrage robuste). Pour répondre à cette problématique, plusieurs verrous scientifiques ont été identifiés et correspondent à trois modules fonctionnels qui s'intègrent dans la structure classique de commande d'une PO (Figure 2-21) :

- La mise en place d'une approche formelle pour développer les filtres de commande tout en respectant un niveau de SIL (*Safety Integrity Level*) donné pour ces fonctions de sécurité.
- L'utilisation d'un module de diagnostic pour déterminer l'état réel de la PO, car le résultat du filtrage est fortement dépendant de la fiabilité de celle-ci.
- La mise en place d'un module de génération des explications à destination d'un opérateur humain qui devra être compréhensible et suffisamment synthétique vis-à-vis de son niveau de connaissance.

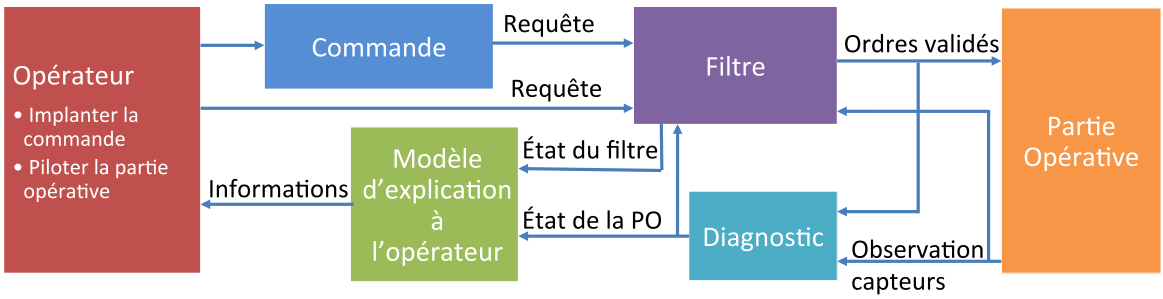

Figure 2-21 : ADEXEC : Introduction d'un filtre de sécurité, d'un module de diagnostic et d'un module d'explication à l'opérateur dans la structure de commande d'une PO

L'utilisation conjointe du filtre de commande et du diagnostic va permettre, d'une part de simplifier le modèle du diagnostiqueur en éliminant les transitions inhibées par la présence du filtre de commande et, d'autre part, d'améliorer la robustesse du filtre qui peut être complété par des informations provenant du diagnostic.

L'ensemble initial de contraintes peut être enrichi en prenant en compte les informations des diagnostiqueurs. Lorsqu'une défaillance survient sur un capteur ou un actionneur, les contraintes du filtre qui contiennent les variables logiques associées aux dispositifs défectueux deviennent erronées. En effet, une évolution de la commande autorisée normalement peut être interdite et, pire encore, une évolution interdite peut alors devenir autorisée. Par conséquent, les contraintes de filtrage doivent tenir compte du fait qu'il y ait une défaillance ou non. Pour chaque partition de défauts, un flag sera mis à vrai lorsque le diagnostiqueur atteindra un état de la défaillance (état sans transition de sortie). Ce flag détermine si la variable considérée peut être utilisée dans la contrainte du filtre (si le flag est faux) ou si une des informations reconstruites doit être utilisée (si le flag est vrai). Dans ce dernier cas, une information temporisée est introduite dans la contrainte à la place de l'information défectueuse. C'est à partir des informations issues du filtre et du diagnostiqueur qu'une explication peut alors être retournée à l'opérateur s'il est nécessaire qu'il intervienne.

Les premiers travaux réalisés ont consisté à cerner de manière exacte le rôle des opérateurs humains. Hormis l'opérateur de conception – celui qui crée la commande qui peut être validée hors ligne – deux « types » d'opérateurs peuvent être distingués :

- un opérateur de production, en lien direct avec la PO et qui peut, en parallèle d'un automatisme, agir sur celui-ci et donc commettre des erreurs,

- un opérateur de maintenance en charge de corriger la PC d'un automate si celle-ci s'avère défaillante.

Les tâches de ces deux types d'opérateurs ne sont pas les mêmes et peuvent conduire à des utilisations différentes du filtre tel qu'il est défini actuellement. Concernant les erreurs issues d'opérateurs de production, on peut prendre le parti pris de donner la priorité au filtre de sécurité, l'impact n'étant au final qu'un arrêt de production. Il reste néanmoins nécessaire de fournir à ces opérateurs suffisamment d'explications pour qu'ils comprennent leurs erreurs et redémarrer la production le plus vite possible. Dans le cas d'une panne d'un élément de la PO, ces opérateurs peuvent aussi enrichir le diagnostic si ce dernier n'est pas capable de déterminer quel élément est en panne. Concernant les opérateurs de maintenance, leur rôle consiste ici à assurer la correction des programmes implémentés sur API, suite au déclenchement du filtre. Ils doivent donc assurer le diagnostic de la commande pour ensuite la corriger.

On comprend bien dans ces travaux que l'Opérateur Humain doit être reconsidéré au sein du système industriel de demain.

3.3. Diagnosticabilité par *model-checking*

Une collaboration avec le CRAN de Nancy autour d'un PAI (Projet Action Incitative) sur la formalisation de la vérification de la diagnosticabilité par *model-checking* a pu également être montée.

L'utilisation d'approches pour le diagnostic des défaillances est indispensable pour les systèmes plus ou moins complexes. Cependant, une question reste en suspend : le système est-il diagnosticable ? En effet, avant d'appliquer une méthode sur un système, il faut pouvoir vérifier si ce dernier dispose d'informations en quantité suffisante pour pouvoir effectuer le diagnostic. Par conséquent, la notion de « diagnosticabilité » va permettre de déterminer l'ensemble des pannes pouvant être diagnostiqué. Selon la structure (centralisée, décentralisée, distribuée) et l'information disponible, des extensions de la notion de diagnosticabilité ont été définies dans la littérature pour les SED.

Un SED est dit diagnosticable pour l'ensemble des partitions de défauts et pour un ensemble d'événements observables s'il est possible de détecter l'occurrence de n'importe

quel défaut appartenant à une des partitions de défaut dans un délai fini.

La notion de diagnosticabilité à base d'événements a été définie formellement dans (Sampath, 1995) :

Définition 1 : Un langage L préfixe clos et vivant est dit diagnosticable par rapport à une fonction de projection PL et un ensemble de partitions de défauts Σ_{Π} ssi :

$$\forall f \in \Pi_{Fi},\ \forall i \in \{1, 2, \ldots, r\}, \exists\, n_i \in \aleph,\ \forall s \in \Psi(\Pi_{Fi}),\ \forall t \in L/s : |t| \geq n_i$$

$$\forall w \in P_L^{-1}(P_L(st)) \Rightarrow f \in w$$

où $L/s = \{t \in \Sigma^* \mid st \in L\}$ l'ensemble de toutes les séquences d'événements après s. $\Psi(\Pi_{Fi})$ est l'ensemble de toutes les séquences d'événements qui se termine par un événement de défaut appartenant à Π_{Fi}. $P_L^{-1}(P_L(st))$ correspond à l'ensemble de toutes les séquences d'événements qui ont une projection, une séquence observable d'événements, équivalente à celle de st.

Cette définition signifie qu'un langage L est diagnosticable si et seulement si, pour toute séquence de défaillance contenant un défaut appartenant à la partition Π_{Fi} des défauts de type Fi, le diagnostiqueur doit être capable d'isoler ce défaut après l'occurrence d'un nombre fini d'événement $n_i = |t|$ et que toute autre séquence w ayant un comportement observable, $P(w)$, équivalent à celui de st, $P(st)$, doit contenir un défaut appartenant à Π_{Fi}. Autrement dit, chaque défaut de l'ensemble des défauts Σ_f doit avoir une signature distincte et observable pour inférer l'occurrence de ce défaut et déterminer son type.

Après la construction du diagnostiqueur G_d et pour que le langage L soit diagnosticable, il suffit que ce diagnostiqueur satisfasse les deux conditions suivantes (Sampath, 1995) :

1) Il existe au moins un état du diagnostiqueur pour lequel le diagnostiqueur décide avec certitude l'occurrence d'un défaut appartenant à une partition Π_{Fi}.
2) Il ne doit pas y avoir de cycles dits « indéterminés » pour lesquels le diagnostiqueur est incapable de décider avec certitude l'occurrence d'un défaut appartenant à une partition Π_{Fi}.

Hormis l'aspect mathématique, l'objectif de ce projet est de proposer une évaluation de la diagnosticabilité par *model-checking*. Cette approche consiste à regarder dans un premier temps la dépendance des modèles locaux du système afin d'établir une distribution du

diagnostic. Un *model-checker* est ensuite utilisé afin de vérifier un certain nombre de propriétés sur l'atteignabilité d'états défaillants nous permettant ainsi d'évaluer la diagnosticabilité des modèles proposés. Ceci s'effectue dans un premier temps localement, puis modulairement et enfin globalement sur tout le système.

La vérification de la diagnosticabilité s'effectue suivant l'organigramme de la Figure 2-22. Chaque diagnostiqueur local, ne dépendant que de ses événements de PO, est évalué pour la diagnosticabilité local du composant. Si l'ensemble des défauts du composant ne peuvent pas être diagnostiqué, alors le diagnostiqueur local est enrichi par des événements externes au composant et la diagnosticabilité modulaire est alors vérifiée. Si la vérification de la propriété n'est pas satisfaite, alors la conclusion est que le système est non diagnosticable pour les défauts testés. Sinon, l'algorithme va vérifier la diagnosticabilité globale du système.

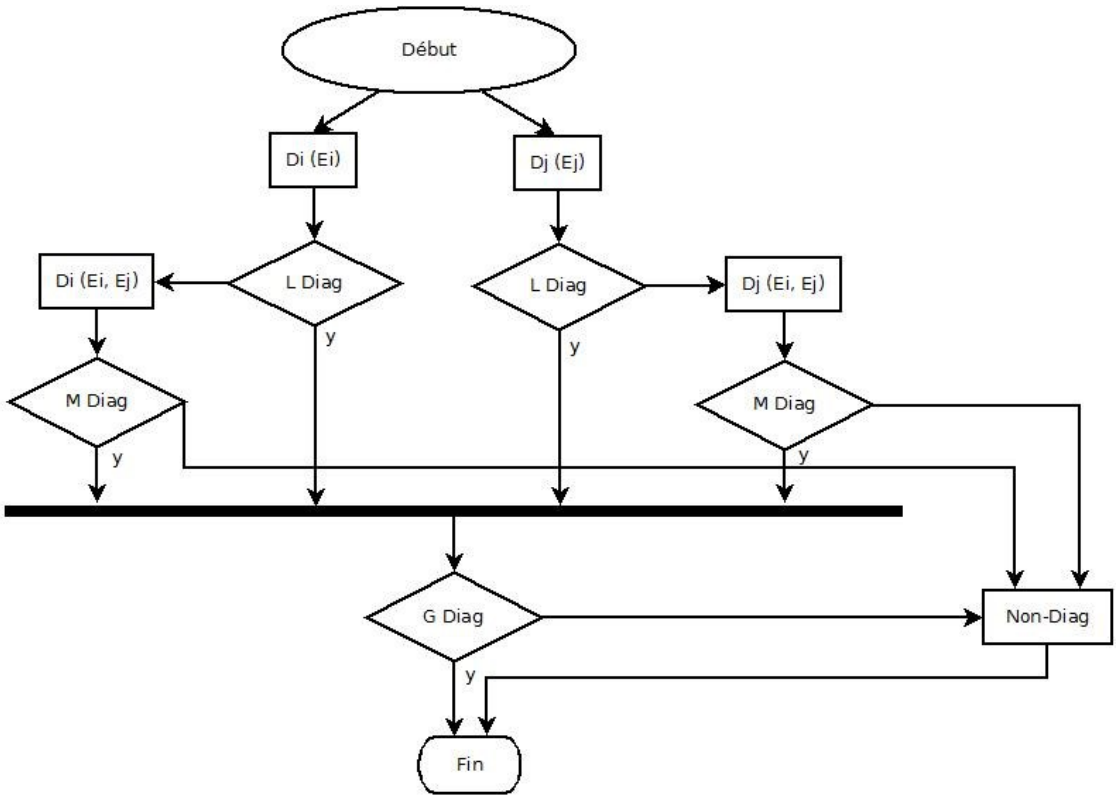

Figure 2-22 : Graphe de flux de la vérification de la diagnosticabilité

Cette collaboration a notamment pu être poursuivie par l'intermédiaire de la thèse de M. Chankate actuellement en seconde année. Quand la plupart des approches de la littérature mettent en avant des algorithmes de résolution performants dans la mesure où le système reste de taille raisonnable et que la partition de faute est très faible (souvent une seule faute

considérée), le travail proposé cible les systèmes complexes possédant de multiples défauts. En effet, outre le problème d'explosion combinatoire de l'espace d'états que l'on pourrait retrouver sur le diagnostiqueur du système, la complexité combinatoire de l'algorithme d'évaluation en est un. Par ailleurs, les différentes approches font très souvent appel à des modèles intermédiaires pouvant rendre la résolution parfois complexe. L'approche présentée dans cette thèse se base sur une méthode de vérification de la diagnosticabilité à travers l'utilisation du *model-checking*. Cette proposition permet notamment l'analyse de la diagnosticabilité à partir du système global du système sans construction de modèles intermédiaires (Marangé *et al.*, 2015).

Une condition de la diagnosticabilité définit dans (Sampath, 1995) étant la présence d'un cycle indéterminé dans le système G, le travail propose de rechercher l'ensemble des cycles (avec et sans faute) et de les comparer entre eux. Dans cette comparaison, il faut alors analyser à la fois si le cycle possède une faute dans sa séquence, mais aussi si celui-ci est distinguable des autres avec certitude. La Figure 2-23 illustre les différentes étapes de la méthodologie proposée :

- Modélisation et Instanciation du système
- Recherche de deux cycles
- Analyse de la non-diagnosticabilité

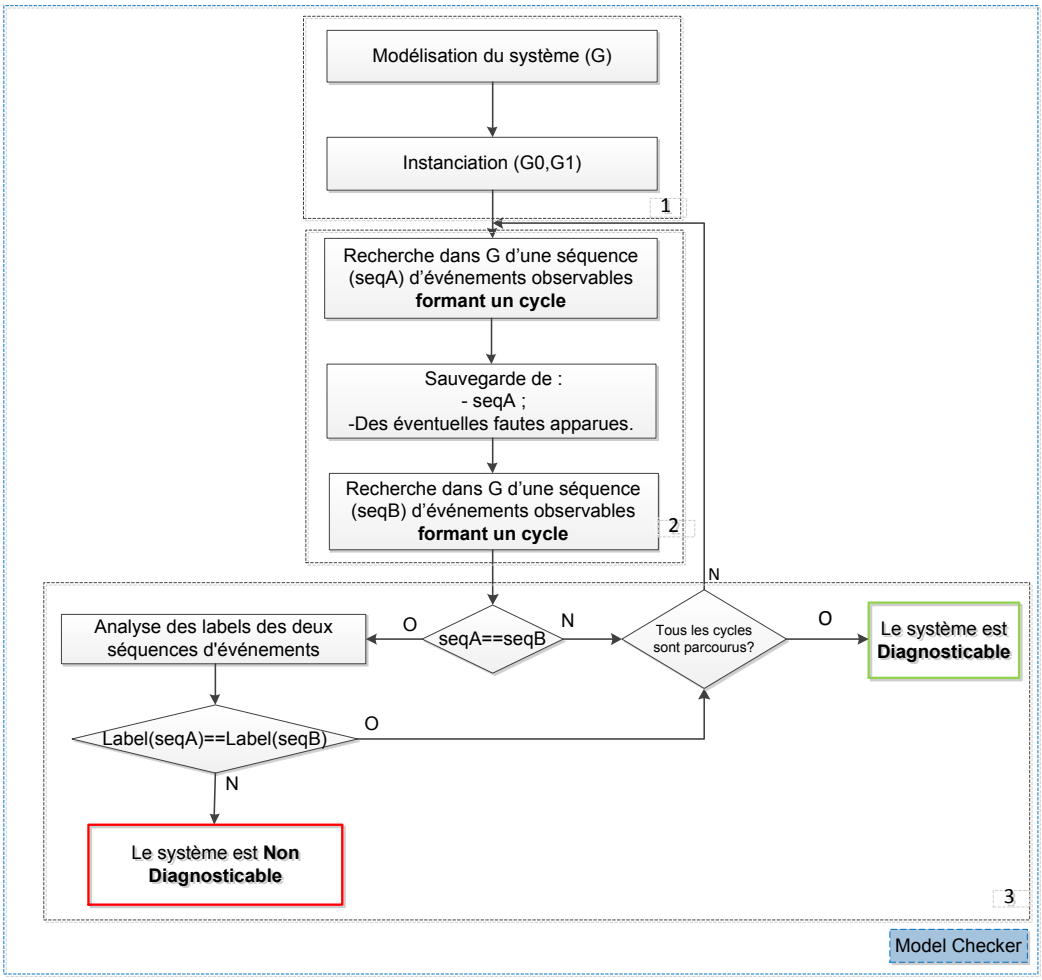

Figure 2-23 : Logigramme de l'approche par model-checking

Que ce soit lors de la phase de vérification de la diagnosticabilité, lors de la conception des diagnostiqueurs ou dans la phase de prise de décision en ligne, l'opérateur doit pouvoir être aidé dans ses différentes étapes.

4. Conclusion du chapitre

Ce chapitre reprend une partie des travaux menés depuis le début de ma carrière. Il y montre un intérêt pour la problématique de la commande et du diagnostic des Systèmes à Evénements Discrets et ce, au travers de diverses propositions duales ou complémentaires. Ces travaux s'appuient notamment sur une phase primordiale : la modélisation. Le modèle est une abstraction mathématique ou graphique d'un système réel qui permet de s'appuyer sur des méthodologies ou de manipuler des algorithmes. Les approches scientifiques interdisciplinaires comme l'Ingénierie Système sont peut-être des solutions pour la

formalisation et l'appréhension de conception de systèmes.

Le modèle répond très souvent à la question du « Pour quoi ? ». Cependant, il convient de ne pas oublier la question du « Pour qui ? ». En effet, que ce soient les systèmes automatisés de productions ou les systèmes de transport, ils cohabitent avec l'Homme. Ils se doivent de l'aider dans ses tâches en vue d'optimiser sa charge de travail. Le modèle s'adresse donc également au concepteur, utilisateur ou même chercheur. L'Homme est utilisateur mais aussi vecteur d'information. Sa place dans l'industrie du futur n'est pas négligeable et doit être étudiée. Il est nécessaire de l'accompagner dans cette transition numérique.

Outre le développement de méthodologie à base de modèles pour la conception de commande et le diagnostic des SED, c'est dans une optique de service à l'opérateur que se situe le chapitre de mon projet de recherche.

Chapitre 3 : L'Opérateur Humain au cœur de l'Industrie du Futur

Ce chapitre présente le projet de recherche que je souhaite entreprendre dans la suite de ma carrière. Il débute par deux constats. Le premier constat est lié aux travaux effectués dans les deux thèses CIFRE (R. Coupat et M. Niang) que j'ai co-encadrées. Au-delà des aspects scientifique, technique et technologique, chacune de ces thèses montre l'importance et l'impact des technologies sur l'Opérateur Humain. Le second constat est quant à lui lié à mes activités de chargé de mission « Industrie du Futur ». En effet, l'usine du futur est mise en exergue un peu partout et pour tout mais nécessite de prendre un peu de hauteur. J'ai ainsi entamé une réflexion sur les changements économiques et sociétaux engendrés par toutes les activités autour de l'industrie du futur.

La Figure 3-1 illustre mon projet de recherche qui est de développer les aspects humains dans mes travaux actuels. On y retrouve la partie constat qui vient d'être évoquée, mes activités de recherche Commande et Diagnostic et l'activité de Formation. Pour appliquer et valider ces différentes réflexions, le projet de plateformes interconnectées « *Factories of Future Champagne-Ardenne* » (FFCA) viendra en appui. Ce projet, répondant au CPER 2018-2020, vise notamment à replacer l'Homme au cœur de l'usine du futur.

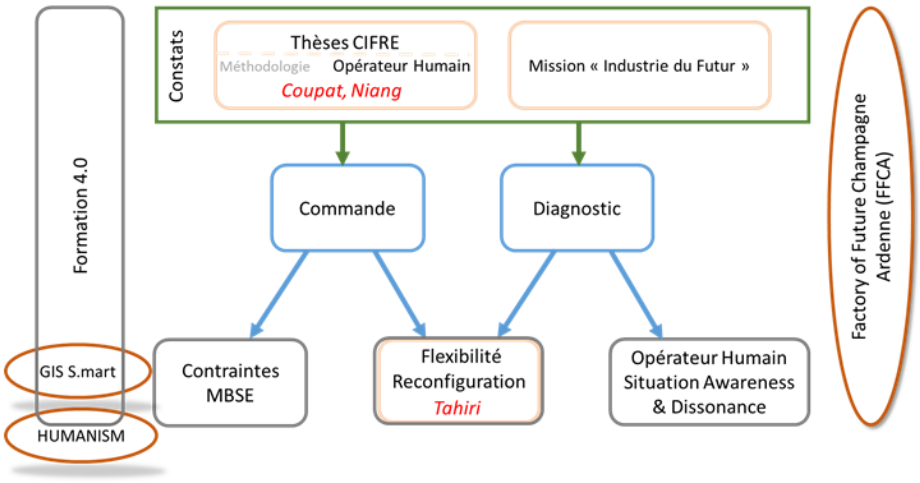

Figure 3-1 : Positionnement du projet de recherche

Le développement de l'aspect Commande passe par le développement d'une aide à l'établissement des contraintes. Pour cela, j'envisage de m'appuyer sur les techniques en ingénierie des systèmes basée sur les modèles (MBSE). Je développe ensuite une deuxième piste de recherche concernant les explications à donner à l'utilisateur suite au module de diagnostic. L'utilisation des concepts de *Situation Awareness* et *Dissonance* issus des sciences humaines est envisagée dans ce cadre. Une troisième perspective de travail concerne la flexibilité des systèmes. C'est un travail déjà entamé avec la thèse de Imane Tahiri autour de la reconfiguration de la commande des systèmes mais qui doit être approfondi avec les principes de reconfiguration des équipements et mêmes des personnes. Enfin, la quatrième piste repose à la fois sur un aspect recherche mais également pédagogique. En effet, l'évolution de nos systèmes et leur concept doivent être accompagnés d'une « Formation 4.0 » pour les utilisateurs que ce soit dans le monde académique ou industriel. Ce travail fait notamment l'objet de deux projets en cours (GIS S.mart et ANR HUMANISM).

1. Constat sur la place de l'Opérateur Humain dans les travaux effectués dans des thèses CIFRE

Cette section reprend quelques travaux de doctorants en relation avec l'Opérateur Humain. Ces thèses CIFRE, a priori devant répondre à des problématiques techniques, ont permis de montrer l'impact de la technologie sur l'Homme et son environnement.

1.1. Amélioration des conditions de travail des chargés d'étude SNCF

Dans le travail de recherche de (Coupat, 2014), les conditions de travail des chargés d'études du domaine des Équipements d'Alimentation des Lignes Electrifiées (EALE) ont été évaluées. Ces travaux ont montré que l'optimisation de la production des études d'électrification nécessite de prendre en considération l'aspect économique, technique mais également humain. Un des axes d'amélioration vise notamment à augmenter la productivité tout en améliorant les conditions de travail des chargés d'études. Pour cela, le premier axe retenu a été la génération automatique des livrables (documentation, schémas, cahier de recettes…) et plus particulièrement du code API réalisés par les chargés d'études. C'est un moyen de réduire le temps de travail des chargés d'études en leur évitant des tâches souvent routinières. Cette amélioration des conditions de travail devrait entrainer une régulation et une

amélioration de leur charge de travail mentale en évitant les erreurs consécutives à la sous-charge et à des pics de surcharge. Toutefois, cela entraîne une modification de leurs activités qui doit être acceptable et acceptée. Ce travail s'est présenté dans une conjoncture favorable pour être accepté par les chargés d'études étant donné leur surcharge de travail généralisée liée au nombre important de projets prévus et au peu de recrutements. Les chargés d'études ont vu le projet comme une aide et non comme un risque ou un concurrent pouvant conduire à la suppression de leur poste. Il est également important de noter que la génération automatique ne peut pas se faire sans standardisation. C'est une condition nécessaire à la génération automatique des livrables et elle ne peut se faire sans le partage du savoir-faire des chargés d'études.

Dans (Debernard *et al.*, 2013), les auteurs se posent la question de « Pourquoi et quand automatiser ? ». Ils définissent 4 vues répondant à cette question : technique, humaniste, économiste et automatique « humaine ». L'approche centrée technique considère qu'il faut automatiser à partir du moment où cela est possible. L'approche humaniste consiste à automatiser les tâches pénibles, ennuyeuses ou encore risquées pour un être humain. L'approche économiste préconise l'automatisation lorsqu'elle revient moins chère. Enfin, l'approche de l'automatique « humaine » considère que l'automatisation ne consiste pas à retirer purement et simplement l'homme de la boucle de contrôle/commande. De ce fait, automatiser consiste à concevoir un système qui accomplit partiellement ou totalement une fonction qui était ou pourrait être partiellement ou totalement réalisée par un opérateur humain (Parasuraman *et al.*, 2000). Cette définition implique que l'automatisation peut varier selon différents niveaux, allant du mode manuel au tout automatique, et qu'un système automatisé est dans bien des cas amené à fonctionner avec un ou plusieurs opérateurs humains. La prise en compte de ces derniers impose donc une approche différente de l'automatisation (Riera *et al.*, 2003) (Debernard *et al.*, 2013).

Dans (Riera, 2001), l'approche de l'automatique humaine a été présentée en l'articulant autour de 4 points :

1- L'objectif principal est l'amélioration globale du Système Homme-Machine (SHM) étudié. Dans le cas des EALE, cela signifie que les solutions ou outils retenus doivent nécessairement être adaptés aux chargés d'études et donc à leur langage métier.

2- Il ne faut pas s'intéresser uniquement aux performances techniques, mais aussi aux effets induits par la mise en place d'un artefact. Pour cela, le SHM doit être étudié en tenant compte aussi bien des caractéristiques de l'Homme (nécessité de modèles décisionnels et d'activités) que de celles du système technique. Le chargé d'études reste dans la boucle de conception du contrôle/commande. Il n'est pas possible ni envisageable aujourd'hui de le retirer. Les outils mis à sa disposition doivent être adaptés à ses besoins et non l'inverse.

3- L'automatique humaine propose des alternatives à l'automatisation centrée sur la technique. La coopération homme-machine (Millot, 2013) et la répartition dynamique de tâches (Debernard *et al.*, 2014) sont des solutions originales proposées et évaluées par l'automaticien humain.

4- L'automatique humaine attache beaucoup d'importance à la phase d'évaluation du système homme-machine. Les critères d'évaluation doivent tenir compte des performances globales du SHM, exprimées par exemple en termes d'écart entre la production réelle et les objectifs, mais également de critères permettant d'évaluer les difficultés rencontrées par les opérateurs humains dans la résolution de problèmes et la communication avec l'outil technique, au travers par exemple de l'évaluation de la charge de travail. Il est primordial d'avoir un retour d'expériences des chargés d'études suite à l'introduction de nouveaux outils et méthodes modifiant ses activités et ses conditions de travail (Vanderhaegen and Caulier, 2011.

L'automatisation des EALE est confrontée à la même difficulté que l'automatisation des systèmes manufacturiers. La qualité des études doit être améliorée tout en ayant un coût moindre. Il s'agit donc de réduire le temps d'étude tout en assurant la sûreté du fonctionnement et la sécurité du système. L'automatisation des EALE doit donc répondre à des objectifs économiques et techniques. Les objectifs économiques s'intéressent aux bénéfices financiers de l'entreprise et portent sur la diminution des coûts de fonctionnement, gain de temps sur les études, amélioration structurelle du système, recherche de solutions économes, … Les objectifs techniques sont liés au processus de conception de la Partie Commande. Pour les atteindre, il est nécessaire de travailler sur :

- La diminution de la durée du cycle d'étude en prenant en compte les durées administratives, d'approvisionnement, de réalisation…

- L'augmentation de la qualité des programmes en assurant le niveau de qualité attendu par les exigences en termes de fiabilité, disponibilité, maintenabilité et sécurité (FDMS),
- L'augmentation de la disponibilité des EALE en diminuant entre autres les temps de maintenance.

Du côté de la composante humaine, les études des EALE et les essais contiennent aujourd'hui des tâches stressantes, répétitives, pénibles pour les chargés d'études. Il est nécessaire d'analyser leur activité au travers du concept de la charge de travail utilisé dans l'analyse des Systèmes Homme-Machine. Il s'agit d'une notion complexe dont l'usage a été étendu à plusieurs secteurs de la psychologie et de l'ergonomie (Richard, 1996). Elle désigne les conséquences physiques et mentales de l'exécution de la tâche sur l'opérateur. La tâche proprement dite et ses contraintes sont regroupées sous la dénomination d'exigences du travail (Leplat, 1997). L'étude de la charge de travail met en relation l'aspect physique et l'aspect mental de l'opérateur, car son état physique conditionne ses capacités mentales et sa tolérance à l'effort (Cnockaert and Floru, 1991).

Le travail de conception et d'étude des EALE, réalisé par les chargés d'études, nécessite en moyenne une forte activité intellectuelle pendant toute la durée du projet et une concentration pour ne pas commettre d'erreurs. Cette concentration peut être altérée par différents éléments parmi lesquels on peut citer la pluralité des projets en cours qu'il doit réaliser dans un temps limité. En effet, un nombre d'heures est alloué à chaque tâche d'un projet, en fonction de la complexité de la structure de l'EALE. Le stress dû aux *deadlines* (Sargent and Terry, 2000) peut conduire à une surcharge mentale. Une autre difficulté rencontrée par le chargé d'études concerne la multiplicité des outils informatiques de réalisation des différentes tâches au sein du *workflow* du projet. Cela ne l'aide pas à optimiser son temps de travail. Le changement d'outils peut entrainer une perte d'informations et des erreurs de recopie. A ce sujet, une relation a été montrée entre la multiplication des ressources et le travail de mémorisation (Baddeley, 1990; Wickens, Gordon and Liu, 1998) pouvant mener à une surcharge mentale (Young and Stanton, 2001).

Cette pluralité des outils, indispensable pour fournir les différents livrables (documents, programmes, schémas électriques…), entraine également une saisie multiple des mêmes

informations concernant un même projet. Cette répétitivité d'actions, en plus de pouvoir être une source d'erreurs et une perte de temps, nuit à la concentration ainsi qu'à l'intérêt que le chargé d'études porte à son travail. Sa charge de travail mentale est de ce fait réduite, il s'agit de sous charge mentale (Stanton, Young and McCaulder, 1997). Enfin, les chargés d'études forment une équipe mais travaillent séparément ce qui peut entrainer des différences d'interprétation des principes mis en place et des différences dans les livrables fournis.

Il est donc concevable que la tâche cognitive de conception des EALE puisse être assistée par un outil logiciel afin d'éviter une surcharge de travail, mais également les erreurs liées à la routine (Coupat, 2014). L'homogénéisation des documents et des livrables, et leur génération automatique, semblent être une solution pertinente car elle permet de gagner en performance et en qualité. Elle facilite la lecture d'un projet par l'ensemble des intervenants. Toutefois, ce travail ne peut pas se faire sans standardisation et peut conduire à une modification des activités des chargés d'études qu'il convient d'évaluer.

La Figure 3-2 indique la méthodologie utilisée permettant d'aboutir à un progiciel « métier » destiné aux chargés d'études et générant automatiquement des livrables de qualité (documents, code API sûr de fonctionnement, …). Elle repose sur la standardisation et commence par l'étude du métier, de ces principes, des documents et programmes existants. Au travers de ces documents d'entrées, et par *reverse-engineering* des programmes API, il est possible de définir des standards. Pour cela, il convient d'une part d'uniformiser les principes de programmation et d'autre part, de s'assurer que le code API existant ne présente pas de défaut. La génération automatique repose sur la création de *templates*, qui consiste à partir de règles à enrichir ou adapter des prototypes (ou exemples) prédéfinis de code. Une étape d'analyse qualimétrique du code API existant est donc primordiale pour valider voire améliorer les prototypes de code. En effet, générer le code source interprétable assure la lisibilité et la maintenabilité des programmes API par les chargés d'études. L'analyse structurelle permet une décomposition du système, et la construction de modèles de la Partie Opérative et de l'architecture de la Partie Commande. C'est à partir de ces modèles (qui sont spécifiques à un domaine « métier ») que nous proposons de définir les *templates*. La modélisation de l'architecture matérielle de la Partie Commande est nécessaire afin d'établir les règles de distribution des programmes dans les API. Les *templates* doivent être intégrés dans un environnement progiciel avec une saisie unique des données d'entrée. Celui-ci doit

permettre de générer automatiquement les livrables, au moyen d'une Interface Homme Machine (IHM) adaptée au langage métier des chargés d'études grâce aux modèles de PO et de PC.

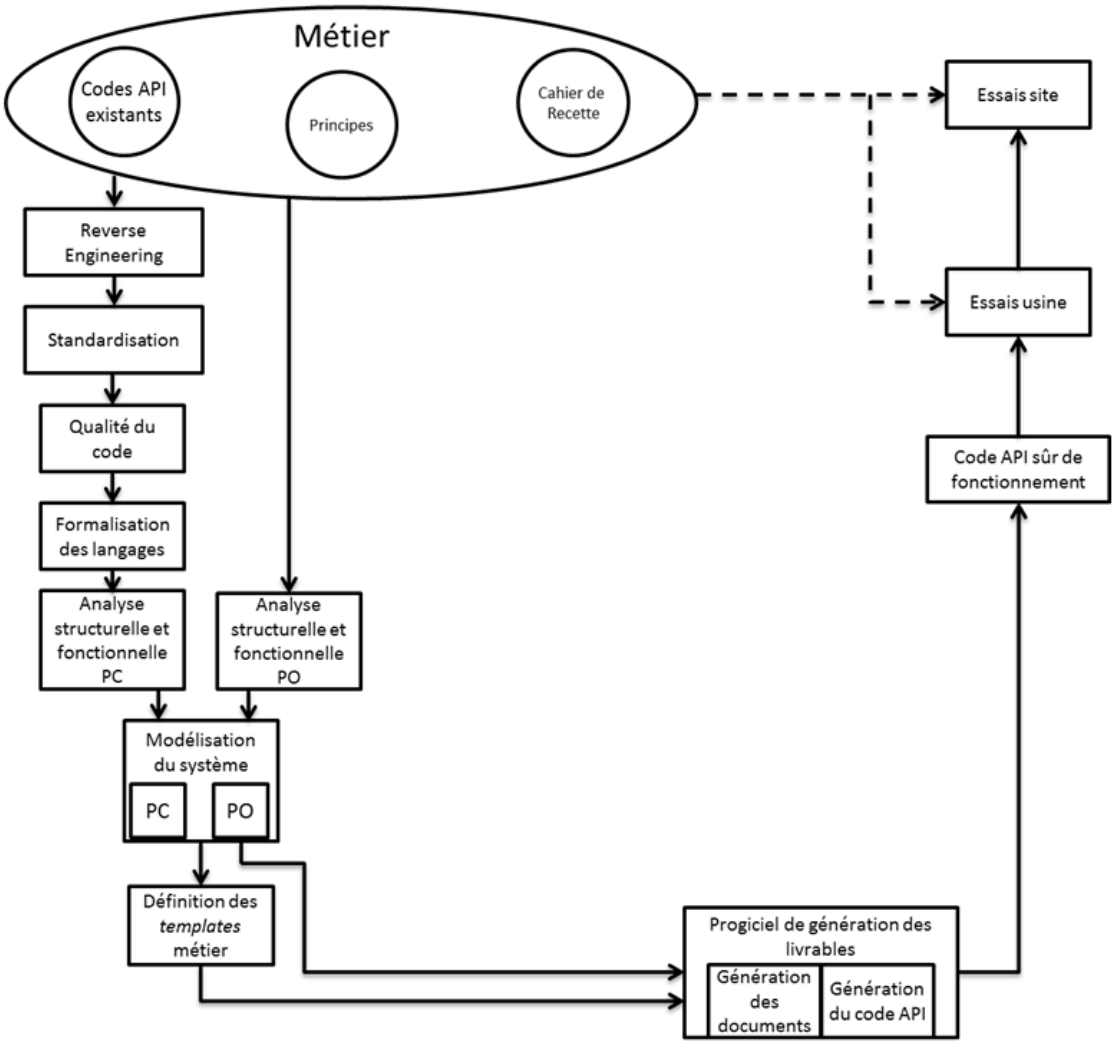

Figure 3-2 : Méthodologie pour les études d'automatisation et la génération automatique de livrables pour la SNCF

Le progiciel Odil GREMLINS a été développé en collaboration avec la société Prosyst. Les modèles et les *templates* ont été intégrés dans Visual Standard. La démarche Agile a été suivie tout au long du développement du progiciel. L'intégration d'Odil GREMLINS a modifié le *workflow* des chargés d'études. Ce nouveau *workflow* supprime les phases de sous charge mentale en les remplaçant par des phases cognitives dont l'enchainement permet d'éviter une surcharge mentale. La saisie unique dans un environnement logiciel unique permet au chargé d'études de rester concentré et de ne pas perdre en performance. Les tâches

de rédaction sont remplacées par la description graphique dans Odil GREMLINS et la vérification des livrables générés. Les chargés d'études peuvent consacrer davantage de temps aux tâches complexes comme la validation des schémas. Les tâches répétitives comme la rédaction du cahier de recettes sont remplacées par des tâches de relecture et de correction si nécessaire, ce qui évite des étapes de sous charge mentale et réduit le temps d'étude nécessaire pour un projet d'électrification.

La charge de travail des chargés d'études peut être diminuée par la mise en place de ce nouveau *workflow*. Etant donné qu'il y a environ 8 projets de régénération de sous-stations par an, ce sont environ 920 heures par an qui peuvent être économisées. Cette démarche a également été appliquée aux postes d'alimentation électrique, dont les régénérations sont des projets plus courts mais plus nombreux sur l'année, environ 20 projets par an. Les attentes des chargés d'études se portent sur les thèmes suivants :

- homogénéité des projets,
- augmentation de la productivité,
- fiabilité des programmes générés et réduction des erreurs,
- réduction du nombre d'outils et utilisabilité de la solution.

Odil GREMLINS a permis de répondre à ces attentes par la génération d'un code standardisé et lisible approuvé par les chargés d'études. Odil GREMLINS est une interface de saisie unique qui permet de diminuer le nombre d'outils, une fois l'architecture de l'installation et du contrôle/commande associé définies, les programmes et les livrables peuvent être générés. La solution permet d'homogénéiser tous les projets par la génération automatique de la répartition des appareils et des informations. L'utilisation de l'outil en autonomie par les chargés d'études ne devrait pas présenter de difficultés particulières. L'outil intègre le langage métier et s'inspire de documents existants (schémas unifilaires) pour la description graphique. Certaines phases de projets peuvent être réduites de 100% par l'utilisation d'Odil GREMLINS sur l'ensemble du *workflow* d'un projet.

Ce travail s'est poursuivi avec la thèse de M. Niang pour l'amélioration des tâches de vérification formelle des programmes API (Niang, 2018).

1.2. Méthodologie de Vérification des systèmes de contrôle/commande à la SNCF

Après les phases d'étude et de conception du système de contrôle commande des EALE, les chargés d'études de la SNCF procèdent à la vérification et la validation du système au moyen de tests de validation en usine (*Factory Acceptance Test* - FAT) puis sur site (*Site Acceptance Test* - SAT) à partir d'un cahier de recettes. Lorsqu'ils réceptionnent les armoires de contrôle commande chez l'intégrateur en usine, ils y transfèrent (après relecture du code API) la totalité des programmes automates qu'ils ont développés. Les procédures de tests vont ensuite être exécutées sur l'ensemble, pour vérifier et valider simultanément les programmes automates et le câblage des armoires.

Les scénarios de tests nécessaires pour vérifier et valider intégralement un système de commande sont très souvent répétitifs et se dénombrent en centaines. Avec des systèmes automatisés aussi complexes que les EALE, ces essais manuels prennent beaucoup de temps aux chargés d'études (1 à 2 semaines). La fatigue générée par ces efforts et le stress lié au *deadline* de fin du projet exposent le chargé d'études à une surcharge mentale qui ne le met pas à l'abri d'erreurs humaines pendant les essais, pouvant lors compromettre la validité des tests réalisés.

Par ailleurs, bien que le cahier de recettes ait servi à vérifier et valider les systèmes de contrôle commande des EALE depuis plusieurs décennies, et que l'expérience ait montré qu'il semblait suffisant pour garantir la sûreté et le bon fonctionnement de l'installation, rien ne prouve que les scénarios de tests sont exhaustifs.

(Niang, 2018) a proposé une approche consistant à vérifier hors ligne, séparément et formellement les blocs fonctionnels qui représentent le code principal du programme automate (Figure 3-3). Le choix d'une approche de vérification par bloc facilite grandement le diagnostic et la correction des erreurs pour des programmes API complexes. Le choix d'une méthode formelle s'impose pour garantir l'exhaustivité de la vérification. Le *model-checking* a été retenu car il répond le mieux aux critères de vérification automatique, langage formel peu complexe, possibilité de génération automatique des modèles...). De plus, son atout majeur réside dans la génération de traces ou contre-exemples après la vérification des programmes, permettant de comprendre l'origine de défauts et de les corriger.

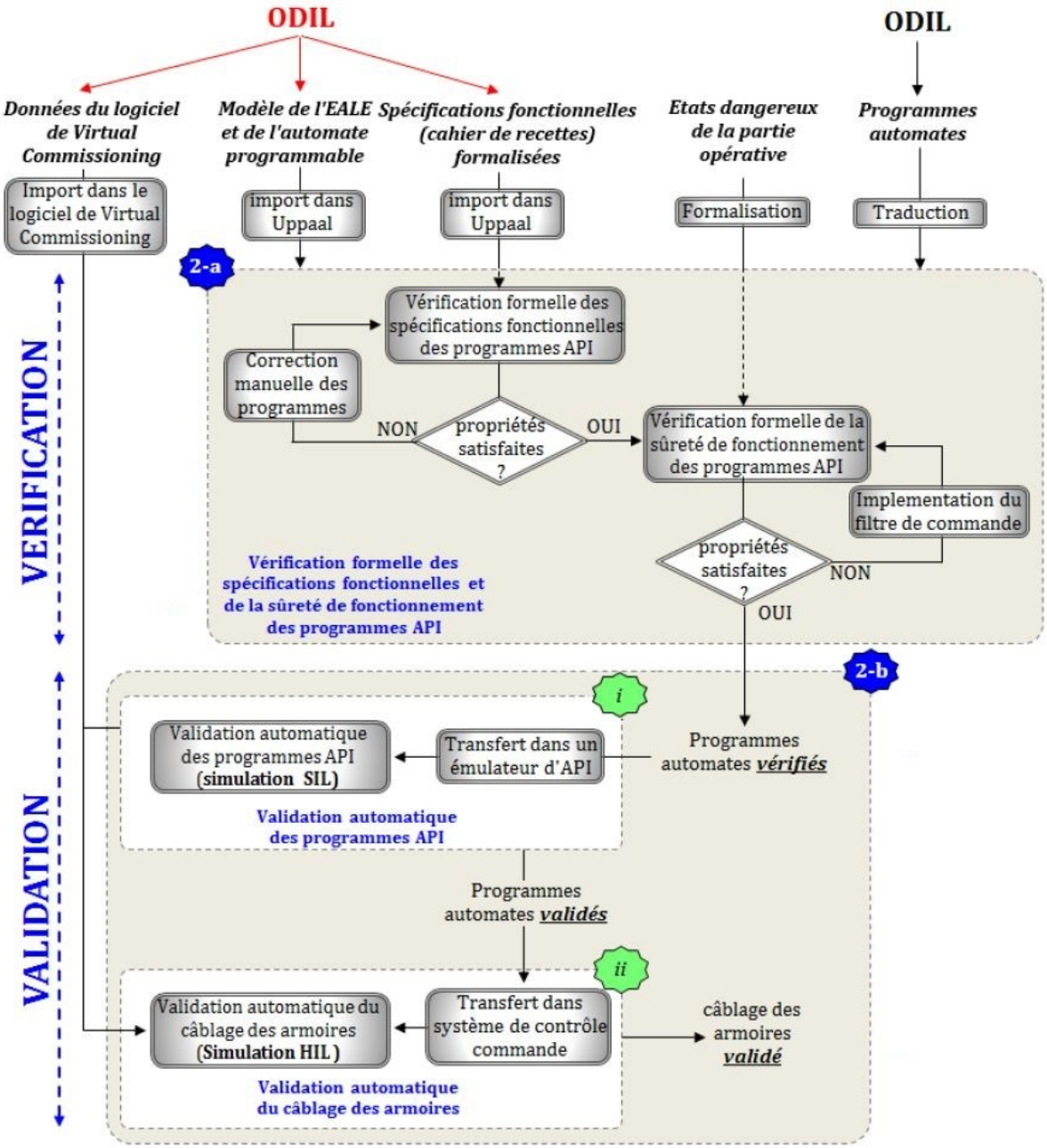

Figure 3-3 : Principe de vérification des programmes API pour la SNCF

La seconde phase consiste à valider directement le programme final grâce à la simulation Software-In-the-Loop. Le principe de cette technique consiste à transférer le programme API dans un émulateur d'automates, puis à faire communiquer celui-ci via une table d'échange avec un logiciel de *virtual commissioning* contenant le modèle virtuel de l'installation électrique et le cahier de recettes numérisé contenant toutes les procédures de tests nécessaires à la validation des programmes API.

A ce stade, les programmes automates sont déjà validés et transférés dans les APIs des armoires de contrôle commande. Mais avant la mise en service, le câblage des armoires doit être validé. Pour cela, le chargé d'études réutilise le même logiciel de simulation précédent (renfermant la PO simulée et le cahier de recettes), sauf que celui-ci sera exploité dans une configuration type Hardware-In-the-Loop, c'est-à-dire connecté au système de contrôle commande réel par l'intermédiaire d'une interface physique.

La validation consiste à exécuter automatiquement toutes les procédures de tests du cahier de recettes sur le système de contrôle commande, ce qui est suffisant pour vérifier que le câblage ne présente pas d'erreurs. Puis après les corrections, le système de contrôle commande est prêt pour la mise en service de l'installation.

C'est en partie suite à ces deux études, aux différents échanges autour des projets réalisés et en cours (MOSYP, ADEXEC, GIS S.mart, ANR HUMANISM) et à la mission sur l'industrie du futur qui m'a été confiée par le Président de l'URCA, que je souhaite recentrer mes recherches sur la place de l'Opérateur Humain au sein de l'Industrie du Futur. Avant de présenter ce projet de recherche, il convient de replacer rapidement le concept de l'industrie du futur et des implications sur l'opérateur humain.

2. L'industrie de demain

2.1. Les concepts et origines de l'Industrie du futur

Le concept industrie 4.0 apparaît pour la première fois lors du salon de la technologie industrielle de Hanovre en 2011. On parle alors de quatrième Révolution Industrielle illustrant une nouvelle façon d'organiser les moyens de production, et ce grâce à l'émergence du numérique et de la robotisation. Elle fait suite à la mécanisation dans l'industrie et l'utilisation de la machine à vapeur au 18ème siècle, à l'électrification et au travail à la chaîne pour la production de masse (fin 19ème, début du 20ème siècle), et à l'apparition des premiers systèmes programmés électroniquement dans les années 1970 (Figure 3-4).

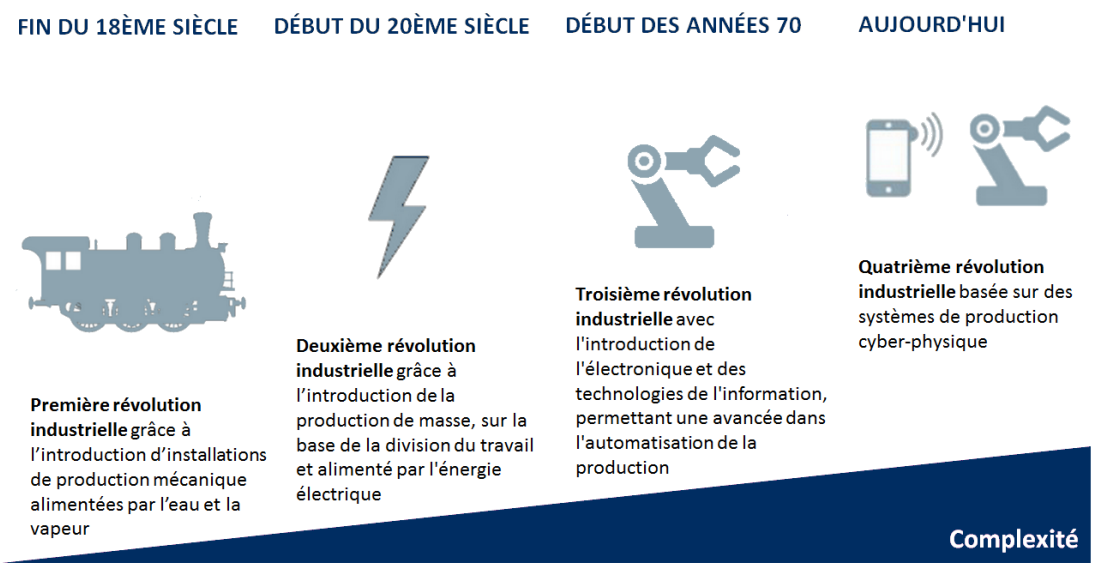

Figure 3-4 : De l'indutrie 1.0 à l'industrie 4.0 (image opi.ch)

Depuis ce salon allemand, chaque pays a élaboré un programme national avec différentes nominations (Figure 3-5). En France, le 12 septembre 2013 un plan gouvernemental « Usine du Futur » est tout d'abord initié, avant d'adopter le nom de « Industrie du Futur » suite à la création de l'Alliance Industrie du Futur (AIF) en juillet 2015 par 11 organisations professionnelles de l'industrie et du numérique, établissements académiques et technologiques (http://www.industrie-dufutur.org/). L'AIF a notamment pour rôle d'accompagner les entreprises françaises dans la modernisation de leurs outils industriels et la transformation de leurs modèles économiques par les technologies nouvelles.

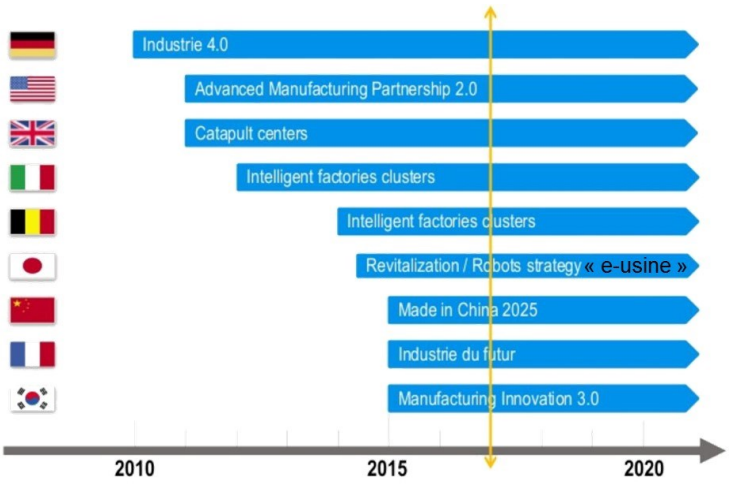

Figure 3-5 : Programmes nationaux (Source : Max Blanchet - société Roland Berger)

Cette quatrième révolution est liée à l'arrivée des systèmes de production cyber-physiques (*Cyber-Physical Production System* - CPPS) qui intègrent l'électronique, le logiciel et la communication, des objets connectés industriels (*Industrial Internet of Things* – IIoT) et du big data. La coordination s'applique non seulement dans l'usine, mais aussi entre les usines afin d'augmenter l'efficacité des processus industriels avec le moins d'intervention humaine tant au niveau de la maintenance que de la gestion des pannes et ainsi, par la flexibilité et la personnalisation, également afin d'accroitre la productivité en réduisant les coûts et la consommation énergétique. Ce n'est donc pas uniquement des aspects technologiques et économiques qui sont impactés, mais également sociétaux et environnementaux. En effet, se posent des questions sur les futurs métiers dans notre société, l'impact sur le chômage, la formation et l'accompagnement des salariés et donc généralement sur la place de l'humain dans cette industrie du futur.

L'industrie 4.0 veut se caractériser par une communication continue entre les différents acteurs et outils intégrés dans les chaînes de production et d'approvisionnement. L'information en temps réel permet une coordination et une flexibilité beaucoup plus importante. Les machines et outils peuvent être paramétrés quasi-instantanément pour une production adaptée/personnalisée. La flexibilité de l'usine et la personnalisation de la production en petite série deviennent plus facilement réalisables, et cela à moindre coût. Le consommateur est donc introduit dans le processus dès la conception. Ce sont les évolutions futures des technologies d'internet qui vont le mettre en lien direct avec l'usine (Figure 3-6). L'ubiquité et l'interconnectivité d'Internet est par conséquent au cœur de tous ces nouveaux concepts.

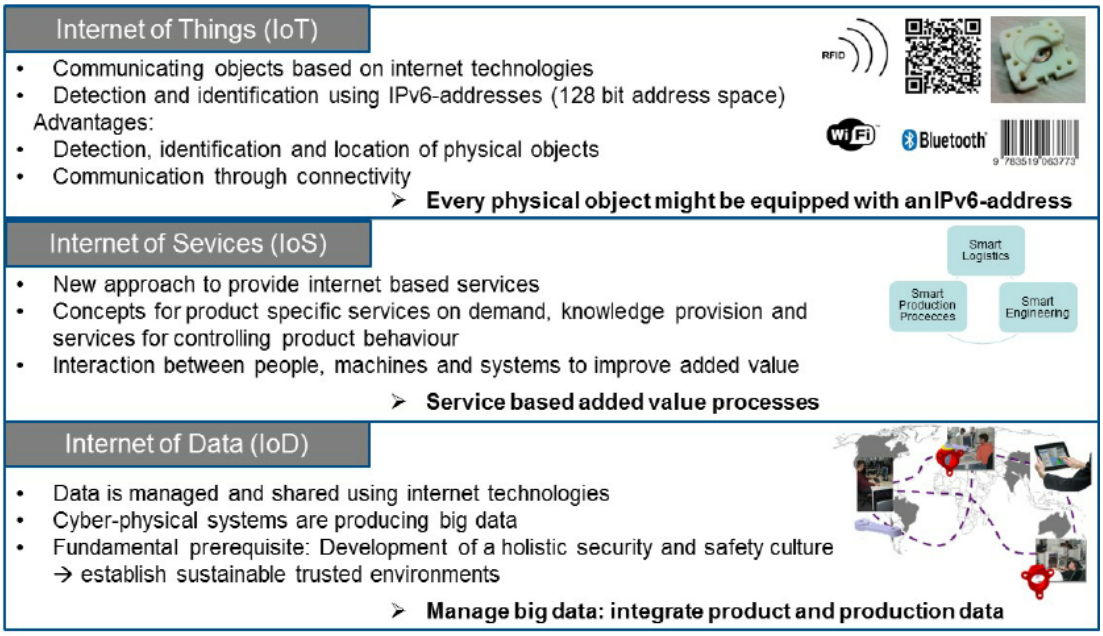

Figure 3-6 : Impact des technologies d'internet (Reiner Anderl. Industrie 4.0 - Advanced Engineering of Smart Products and Smart Production, 2014)

L'industrie du futur génère un flux d'informations devant être échangé le plus rapidement possible que ce soit au niveau de la production ou de la logistique. Le recueil de données sur les différents éléments de la chaîne de production permet d'établir des répliques virtuelles de cette chaîne afin de générer des simulations de procédés ou des tests. Ces outils vont permettre aux futurs ouvriers et techniciens de se familiariser avec les nouveaux outils de travail ou encore faciliter les opérations de maintenance.

Elle doit par ailleurs faire face aux problématiques actuelles de gestion des ressources et de gestion de l'énergie afin de retourner aux utilisateurs des indicateurs concrets de production/consommation.

2.2. Aspects économiques, sociaux et sociétaux

Dans une société moderne, respectueuse de l'environnement, les économies d'énergie et la réduction d'émissions de gaz à effets de serre sont deux objectifs importants dans différents secteurs d'activité. Il s'agit d'une démarche d'amélioration continue qui vise à réduire l'empreinte économique, écologique et sociale d'un produit ou d'un service au travers d'un moyen de production plus intelligent, les *Smart Factories*. En conséquence, que ce soit au niveau de l'innovation, de l'ingénierie ou de la production, il est nécessaire de voir l'industrie comme un « Système de Systèmes » qui est un concept de l'ingénierie des systèmes.

La société se retrouve impactée par ces nouveaux modes de production. La robotisation/automatisation des ateliers, leur flexibilité et leur interopérabilité impliquent une réorganisation du travail dans le temps et dans l'espace. Un conflit se crée entre la mondialisation du processus et l'acceptabilité d'une situation locale. Le phénomène n'est pas inédit. Chaque siècle apporte son lot d'innovations, condamnant des métiers, en créant d'autres. L'économiste J. Schumpeter déjà au siècle dernier appelait cela la « destruction créatrice ». La progression des technologies peut impliquer une mise au chômage de personnes non qualifiées ou ne pouvant être accompagnées dans cette transition industrielle. La relation entre l'Homme et les machines doit être repensée car de nouveaux métiers sont en émergence. Au même titre que la « robolution » annoncée par Bruno Bonnell (fondateur d'Infogrames, directeur de Robopolis, président d'Awabot), l'industrie 4.0 offrira aussi de nouveaux emplois et des perspectives de relocalisations de certaines activités économiques. Les métiers de demain n'ont pas encore été inventés mais ces emplois se substitueront-ils en nombre aux emplois détruits en termes de compétences ou de répartition géographique ?

- *« Les robots, le chômage et les emplois de 2030 »* (France Info, 10/05/2015),
- *« Robots au travail : 3 millions d'emplois menacés en France d'ici 2025 »* (La Voix du Nord, 25/05/2016),
- *« Des centaines de milliers d'emplois créés par la robotique »* (Monster, 16/04/2015),
- *« La vérité sur les robots destructeurs d'emplois »* (Slate, 06/06/2016).

Par opposition avec la pyramide CIM (*Computer-Integrated Manufacturing*) des années 1980, la vision induite par l'industrie du futur semble s'orienter vers une plus grande responsabilité dans l'activité de l'opérateur. Celui-ci doit résoudre des problèmes plus importants que la simple mise en œuvre de ses qualifications. Il va mettre en avant son potentiel d'optimisation et de réactivité. On parle alors de travail participatif. Les tâches y seront plus variées en raison des changements de gamme de produit à fabriquer. L'ouvrier de demain sera plus flexible afin d'acquérir une plus grande autonomie. Il doit donc être accompagné grâce à la formation.

Il est donc nécessaire de mettre en place des stratégies de formation appropriées afin d'accompagner les opérateurs dans cette transition tout au long de leur carrière professionnelle. L'interaction « Homme-Machine », la conception d'interfaces utilisateurs (IHM), doit permettre de trouver des approches les plus adéquates pour que l'entreprise

réussisse à reconfigurer la chaîne de valeur. D'une manière générale, les grands groupes comme les PME doivent faire d'importants investissements matériels et humains afin d'intégrer ces nouvelles technologies numériques.

Malgré tous les avantages annoncés par l'industrie du futur, elle peut se heurter à des limitations techniques et engendrer des bouleversements au niveau sociétal. Elle nécessite la coopération de toutes les entités de l'entreprise mais aussi des périphériques. Par ailleurs, la recherche de standardisation de plates-formes ou d'outils peut constituer un frein à son développement. En effet, il est facile de penser qu'une trop grande homogénéité risque d'impacter la créativité ou que les grandes entreprises risquent d'étouffer le marché. Par ailleurs, la problématique de la cyber-sécurité est devenue inhérente à tout développement. La standardisation dans les réseaux de communication est donc un point clé de la digitalisation de l'usine. Dans ce cadre, OPC UA (*Unified Architecture*) apparaît comme le nouveau standard qui devra être mis en place dans l'industrie.

Les trois premières révolutions ont créé le paysage industriel que nous connaissons aujourd'hui et ont été vecteurs de progrès sociaux et humains. Les progrès technologiques ont permis d'augmenter la productivité des usines et la création de nouveaux emplois. Dans cette quatrième révolution, où se situe la limite ? Sera-t-elle technologique ou humaine ? Quelle sera donc la place de l'homme dans l'industrie du futur ?

En lien avec le constat lié suite aux travaux en CIFRE avec la SNCF et aux motivations affichées de l'industrie du futur, la section suivante présente quatre réflexions de recherche.

3. Commande et Diagnostic : un modèle pour qui ?

Mes précédents travaux ont montré un lien fort entre la synthèse d'une commande sûre de fonctionnement et le diagnostic de défaillances. Cependant, les approches se sont reposées sur un modèle répondant plus au « pour quoi ? » que au « pour qui ? ». Dans la suite de ce chapitre je vais essayer de montrer que des perspectives de recherche sont à envisager dans ce domaine, que ce soit pour la commande ou le diagnostic des SED.

3.1. L'Ingénierie Système pour l'obtention de contraintes

Les différents travaux à base de SCT ou de filtre logique présentés au chapitre 2 ont montré qu'une des principales difficultés réside dans l'élaboration des contraintes qu'elles soient de sécurité ou de vivacité fonctionnelle. En effet, un cahier des charges est rarement complet et ne permet pas toujours d'établir l'ensemble suffisant de contraintes. Ce travail est souvent issu d'une expertise humaine et donc faillible. De plus, lorsque plusieurs dizaines voire centaines de contraintes existent, certaines ne sont peut-être pas nécessaires, ou sont redondantes. Il est donc envisageable de cibler plusieurs points d'amélioration.

Une contrainte peut être incluse dans une autre et donc ne pas être nécessaire, des travaux théoriques sur l'inclusion de contraintes logiques doivent pouvoir être menés. L'objectif est de pouvoir réduire automatiquement un ensemble de contraintes logiques afin d'en simplifier l'analyse.

Point plus intéressant encore : l'aide à la conception des contraintes. Le chapitre 2 a montré que les approches de commande reposent sur l'utilisation de différents modèles : modèle de spécification ou modèle structurel. La modélisation de la Partie Opérative permet notamment d'exprimer tous les comportements possibles d'un système. C'est donc une source d'informations à la fois structurelle mais également fonctionnelle puisque parmi tous les chemins possibles, il existe celui qui est recherché dans un cahier des charges. Les travaux menés ont permis par ailleurs d'identifier que les systèmes industriels complexes sont souvent le résultat d'un assemblage d'éléments simples et normalisés (vérins, capteurs de position, convoyeurs, etc.). Ces modèles d'éléments en interactions, appelés Eléments de Parties Opératives (EPO) ou *Parts of Plant* (PoP) selon leur granularité, peuvent s'assembler de façon hiérarchique afin de reconstituer le système.

C'est sur cette base qu'il me semble possible d'envisager d'aider le concepteur dans l'élaboration d'un ensemble de contraintes. En effet, au même titre que les approches d'ingénierie de systèmes sûrs de fonctionnement basées sur les modèles (Bévan *et al.*, 2012), des modèles conceptuels de haut niveau doivent pouvoir permettre l'interprétation automatique de contraintes. L'idée étant d'essayer de bien modéliser un « tout » dans son environnement et non seulement dans son objectif fonctionnel. Les travaux menés en

ingénierie des systèmes, et notamment ceux sur l'utilisation de méta-modèles (Figure 3-7), illustrent le fait que des liaisons de communications peuvent être représentées depuis la structure du système (*Block diagram, Behavior diagram*). Ces relations fortes en MBSE (*Model-Based System Engineering*) sur du pluridisciplinaire (mécanique, pneumatique, électrique, informatiques…) devraient pouvoir être une base de réflexion à cet ensemble de contraintes (Chapurlat and Bonjour, 2014) (Figure 3-8).

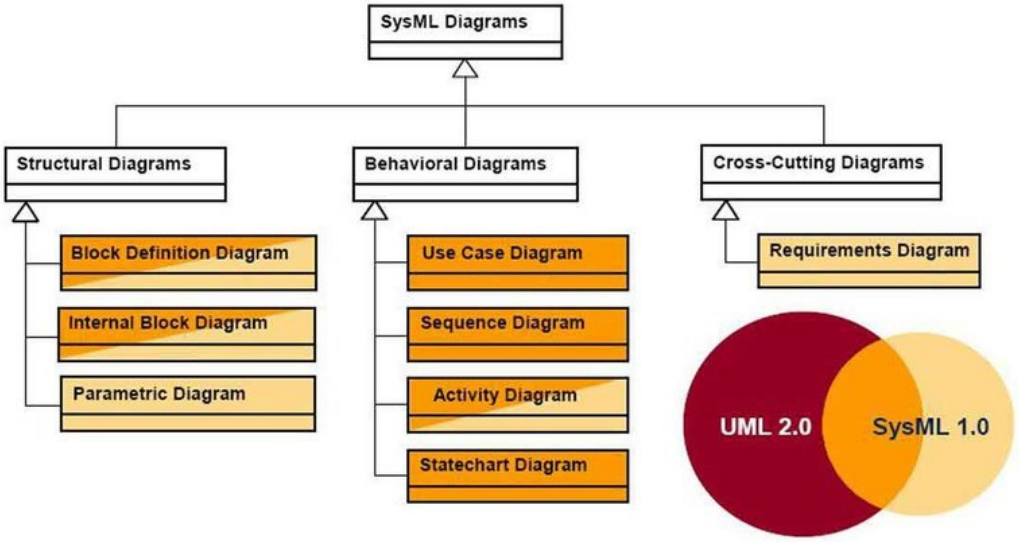

Figure 3-7 : Extrait de la spécification de SysML - Object Management Group, Inc. (C) OMG. 2008.

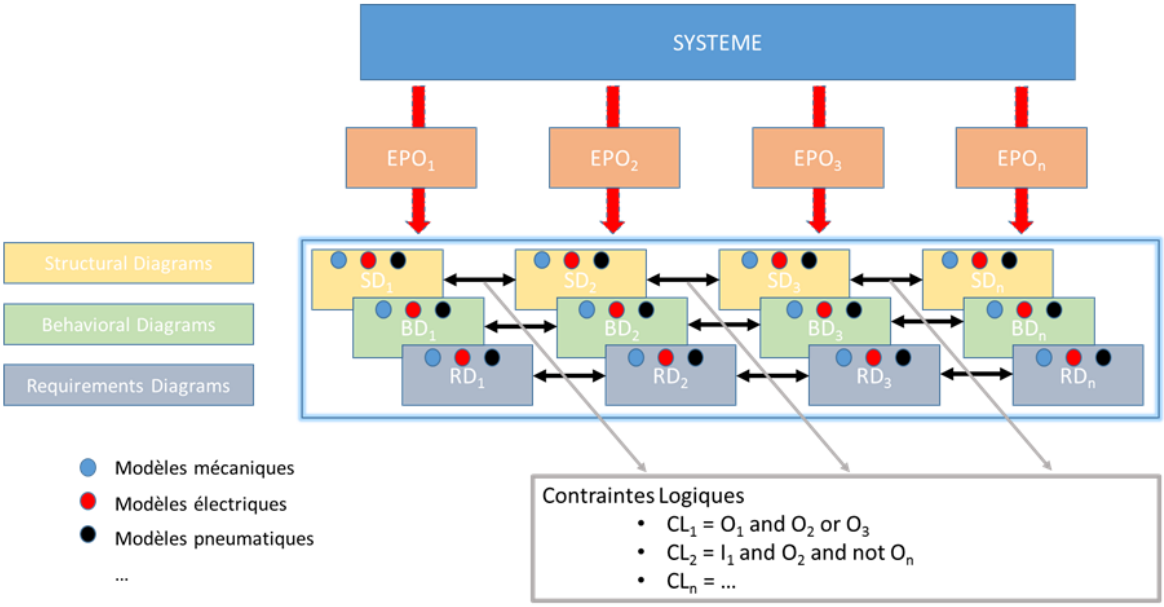

Figure 3-8 : Réflexion sur les interactions de modèles en IS pour la conception de contraintes

Deux points supplémentaires peuvent être ajoutés à cette réflexion.

Premièrement, au regard des concepts de l'industrie du futur, la digitalisation, dans le sens du jumeau numérique (*Digital Twin*), permet de reproduire le plus fidèlement possible les comportements d'un système réel. Ces outils d'émulations numériques sont basés sur des modèles informatiques issus de la CAO (Conception Assistée par Ordinateur) et donc par définition issus de Méta-modèles. Dans le cas de la conception d'un ensemble de contraintes, celles-ci doivent pouvoir être extraites automatiquement à partir de ce modèle numérique. En effet, en simulant le système dans sa globalité, des contraintes logiques devraient pouvoir être inférées.

Deuxièmement, l'industrie du futur impose de nouveaux modèles pour l'entreprise : de la conception à la fabrication. L'ère où les systèmes sont complètement interconnectés et communiquent entre eux de manière intelligente impose une continuité numérique verticale et horizontale au sein de l'entreprise. La mise en place d'OPC UA (Unified Architecture) permet de rendre l'ensemble des systèmes interopérables, tout en garantissant une sécurité des données. Cette brique technologique existe mais nécessite encore des apports scientifiques dans son implantation. En effet, la donnée doit être structurée afin de devenir disponible en temps réel de bout en bout de la chaîne, depuis la conception jusqu'à son démantèlement, principe même du cycle de vie en IS. Ce sujet de recherche fait actuellement l'objet d'une discussion avec un « expert technologie et architecture d'automatisme » du Technocentre Renault Guyancourt.

3.2. Détecter, Isoler mais pas que

Le diagnostic s'intègre dans le cadre plus général de la surveillance et de la supervision d'un processus. C'est un système d'aide à la décision dont l'objectif est de détecter, localiser et identifier les composants ou les organes défaillants. Le diagnostic établit donc un lien de cause à effet entre un symptôme observé et la défaillance qui est survenue, tout en considérant qu'un même symptôme peut apparaître pour différentes causes. Cependant, les approches de diagnostic des SED dans la littérature se focalisent exclusivement sur les phases de détection et de localisation (isolation), mais se trouvent souvent limitées par :

1. La difficulté à garantir l'absence (N), ou au contraire la présence (F), d'un

(plusieurs) défaut(s) avec certitude dans un délai fini. On parle alors d'ambiguïté ou d'incertitude.

2. La difficulté à exprimer les caractéristiques d'un défaut (pourquoi est-il apparu ?, Quel impact a-t-il sur le système ?, etc.).

Les approches identifient le « qui », parfois le « quand » mais trop rarement le « pourquoi » et encore moins le « par conséquent ». Hors, par définition, l'Opérateur Humain (OH) attend de la tâche de diagnostic un outil d'aide de prise de décisions, d'explication à la défaillance et d'expressions des conséquences évitées. Ces informations lui permettent de prendre conscience d'une situation afin d'interagir sur le système. Ce concept de *Situation Awareness* est issu des travaux de supervision dans (Millot, 2013).

Par ailleurs, la perception ou interprétation d'une défaillance par un opérateur peut être différente de celle d'un de ses collègues. Elle peut même être contradictoire avec ce que le module de surveillance lui retourne (Figure 3-9).

Figure 3-9 : Perception vs Supervision

Dans le cadre d'études autour de l'usage de technologies, tel que le diagnostic, sur des systèmes complexes, il est intéressant de regarder les travaux autour de la notion de dissonance cognitive ou *dissonance Engineering* définis dans (Vanderhaegen, 2014). Cette

notion, issue du monde de la musique, exprime une discordance d'un ensemble d'informations impliquant un sentiment de doute, de contrariété (Figure 3-10). Dans le même principe, il paraît évident qu'une explication « floue » ou incomplète d'une situation après défaillance, ne permettra pas à l'OH d'être serein dans son intervention (Vanderhaegen and Carsten, 2017). Il existe un besoin de cohérence et il convient donc :

- soit de compléter les informations issues d'outils de diagnostic par d'autres modèles cognitifs pour rassurer l'OH,
- soit d'évaluer les risques relatifs à des conflits potentiels entre connaissances.

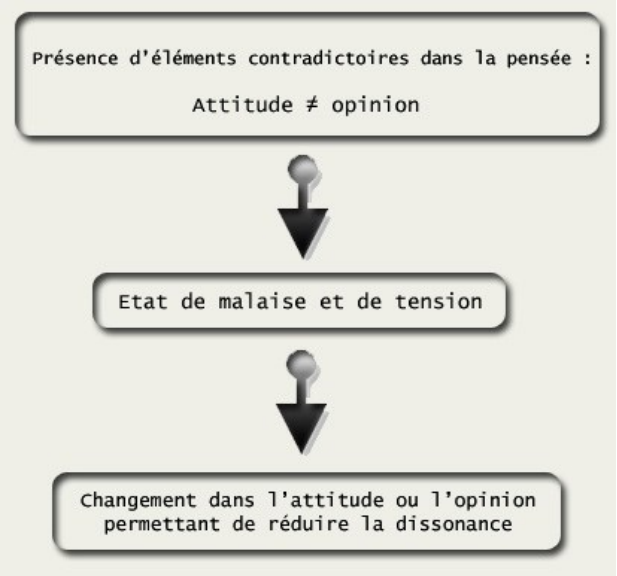

Figure 3-10 : Principe de dissonance cognitive

Attention, cette cohérence d'informations ne doit pas forcément être synonyme de « confort » pour l'opérateur (Figure 3-11).

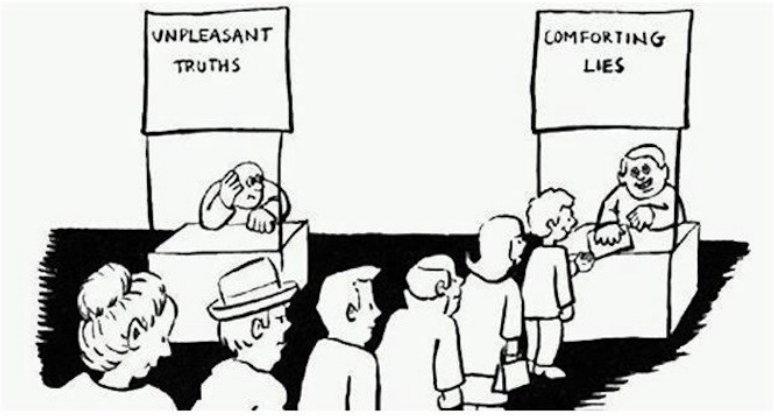

Figure 3-11 : Se rassurer en ignorant

Il est intéressant de remarquer que, malgré que l'on se situe dans une phase de maintenance, on retrouve ici les mêmes problèmes que ceux rencontrés dans les étapes de conception de la commande mais cette fois lors de la phase d'expression du besoin au niveau diagnostic. En effet, il est de nouveau question d'interprétation de l'information non plus donnée par un cahier des charges, mais par un module de diagnostic (Figure 3-12).

Figure 3-12 : L'expression du besoin du diagnostic ?

Sachant qu'une vérité n'est vraie que dans un contexte défini, un point de vue informatique ou humain mériterait d'être discuté autour de sa structure et de sa fonctionnalité. A l'image du tableau de R. Magritte (Figure 3-13), il convient de se questionner et de questionner nos étudiants sur les différents usages des objets qui nous entourent.

Figure 3-13 : René Magritte – La Trahison des images « Ceci n'est pas une pipe »

3.3. Reconfiguration et commande tolérante aux fautes

Les systèmes industriels sont amenés à évoluer, à se complexifier et à s'adapter dynamiquement à l'environnement de travail humain et technique. On parle d'usine du futur flexible dans le sens où tout devient connectable, tout devient simulable, tout devient reconfigurable. C'est dans ce contexte que l'étude de la reconfiguration de lois de commande pour les SED paraît intéressante. Il ne convient plus uniquement de détecter et localiser un défaut, mais aussi de fournir à l'utilisateur des solutions, automatiques ou non, d'interventions/interactions avec sa machine. En maintenance, cela se traduit comme l'ensemble des actions techniques et administratives destinée à maintenir (maintenance préventive) ou à rétablir (maintenance corrective) une entité dans un état spécifié ou dans des conditions données de sûreté de fonctionnement (disponibilité, fiabilité, maintenabilité et sécurité) pour accomplir une fonction requise. Ce processus de réorganisation physique ou logique des postes de travail au niveau implantation ou allocation à des produits et/ou à des activités peut être attribué à la réorganisation de la commande. On parle alors de reconfiguration pour de la commande tolérante aux fautes. Par ailleurs, si l'on se heurte à une problématique de modélisation globale du système dans les approches à base de SCT classique, il en est de même lorsque l'on souhaite étudier la reconfiguration d'un système.

Le processus de reconfiguration du contrôleur consiste à établir une commande capable de s'adapter et d'exploiter les services encore disponibles offerts par la partie opérative (Frizon de Lamotte, 2006). En effet, il est nécessaire d'identifier les différents « services » perdus suite à l'occurrence d'une défaillance. Cette connaissance est généralement issue de la fonction diagnostic qui a pour but de localiser puis d'identifier la cause de la défaillance. Il est à noter ici que le rôle d'une telle fonction diagnostic dans le cadre du processus de reconfiguration n'est pas uniquement d'identifier la cause de la défaillance dans un objectif de maintenance. La fonction de diagnostic doit également identifier la défaillance ainsi que ses conséquences directes ou indirectes sur les capacités de la partie opérative, ceci dans l'objectif de mettre à jour le modèle des capacités opératoires de la partie opérative.

Un sujet de thèse en cotutelle avec l'Université de Cadi Ayyad est actuellement en cours. Cette thèse s'intéresse à la reconfiguration de la commande suite à un défaut process (Figure

3-14). Elle s'articule autour d'un module de diagnostic à base de modèles permettant de prendre une décision de reconfiguration du système en cas de défaillance. Cette reconfiguration est exécutée par un coordinateur qui va faire « switcher » le module de commande (G_{DC}) utilisant l'information corrompue vers un module équivalent tolérant à la faute (G_{TDC}). Dans ce travail, l'architecture retenue pour la commande est distribuée.

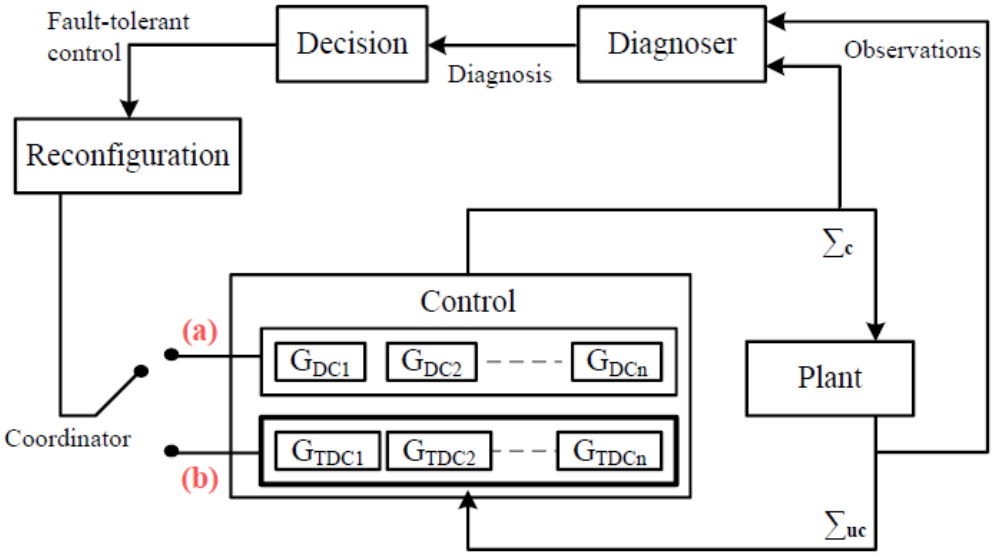

Figure 3-14 : Boucle de Commande Tolérante aux fautes

Cependant, pour assurer la flexibilité de l'usine, il serait intéressant également de regarder les travaux autour de la reconfiguration du process et les réflexions sur la reconfiguration de l'opérateur humain dans ses tâches (Delorme *et al.*, 2019).

3.4. Formation 4.0

L'aspect social évoqué dans l'introduction de ce chapitre montre bien que dans l'industrie du futur les opérateurs 4.0 devront être capables d'intervenir sur des systèmes virtuels et réels, de détecter une situation imprévue et d'interagir, de faire face aux problèmes de qualité et de sécurité des données de production. Ils seront devant une panoplie d'outils et d'approches qui leur faudra maîtriser. Il faut donc accompagner dès maintenant les opérateurs d'aujourd'hui et ceux de demain grâce à la formation. On parle de « formation 4.0 » dans le sens de cette 4^{ième} révolution industrielle ou les opérateurs et formateurs seront en face d'outils réels comme virtuels.

Le projet GIS S.mart (ancien réseau AIP Priméca), tout comme le projet ANR HUMANISM, débutés cette année 2018 et décrits en chapitre 1, visent à développer une pédagogie innovante avec des outils numériques (propres à l'automatique) en interconnexion avec un jumeau numérique, autour d'une plateforme permettant d'avoir une vision systémique multi points de vue du système de production. Le jumeau numérique est un moyen pour l'apprenant d'interagir sans risque physique avec une partie opérative, de confronter le propre modèle de fonctionnement qu'il en a, avec les comportements « réels » de ce jumeau et donc d'améliorer sa compréhension, et de favoriser sa propre « conscience de la situation ». Les avantages attendus sont l'augmentation de la motivation des élèves et donc de son implication, et de favoriser leur apprentissage. Ce jumeau numérique pourra permettre de réaliser du *virtual commissioning* pédagogique intégrant des capacités de coopération (en particulier explicatives) avec l'apprenant. Les travaux autour de la synthèse de la commande par contraintes logiques de sécurité au moyen d'un filtre bloquant, et ceux de la coopération Homme-Machine sont des chemins à étudier. Des scénarios et des cas d'usage avec une pédagogie innovante (situations problème, classe inversée, approche par projet…) pourront être utilisés.

Ces projets sont en partenariat avec des collègues de plusieurs établissements (URCA, Univ. Valenciennes, Univ. Lorraine d'un côté et URCA, Univ. Valenciennes, Univ. Bretagne Sud de l'autre) dans les domaines de la formation au contrôle/commande, la synthèse de la commande sûre de fonctionnement, la coopération Homme-Machine et la simulation de PO.

4. Conclusion du chapitre

Ce troisième chapitre introduit des pistes de réflexion pour des projets de recherche avec notamment un recentrage des activités de Commande et de Diagnostic autour de l'Opérateur Humain (Figure 3-15). Dans un contexte d'usine du futur, que ce soit en pédagogie ou en milieu industriel, il est nécessaire d'appréhender cet aspect dans nos activités de recherche.

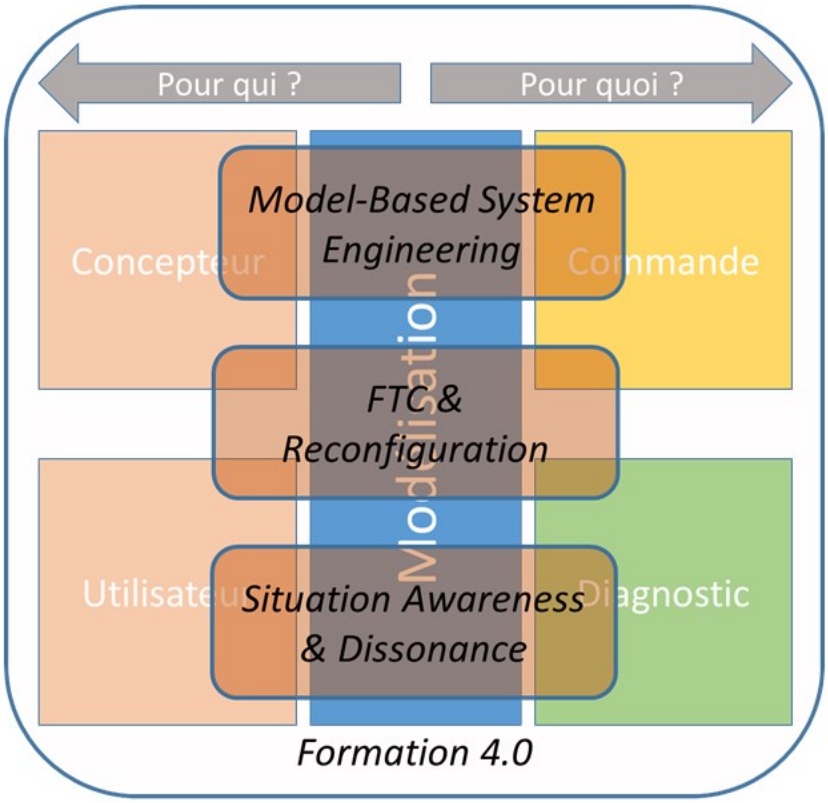

Figure 3-15 : Perspectives de travail

Cependant, si l'on souhaite considérer l'usine du futur comme une thématique clé pour le développement industriel, il convient de s'appuyer sur des ressources et des collaborations entre les industriels et les académiques. Parmi ces collaborations, la création d'une plateforme technologique de pointe permettant d'implémenter, de tester et de valider les méthodes et les approches développées dans ce cadre. Cette étape est particulièrement nécessaire avant l'exploitation définitive des approches en industrie.

Dans le cadre du projet de plateformes interconnectées *Factories of Future Champagne-Ardenne* visant à replacer l'Homme au cœur de l'usine du futur et en particulier de sa construction, l'un des points clés est l'ouverture de la technologie à l'Homme. Ce projet, du CPER 2018-2020, est un investissement important au niveau de la région Grand-Est pour une meilleure visibilité de l'ex-région Champagne-Ardenne sur ces problématiques.

La plateforme sera organisée de façon à être un lieu d'accueil et de diffusion orienté vers le monde de l'éducation et de la formation. Elle sera accessible aux plus jeunes, en vue de les sensibiliser aux principes qui sous-tendent ces évolutions technologiques et sociétales, mais aussi aux formateurs et aux salariés des entreprises dans le cadre de la formation continue.

Cette plateforme devra permettre de réaliser des programmes de recherche portés par les laboratoires, les centres techniques et les entreprises. De nouvelles approches pédagogiques et de nouveaux formats d'apprentissages arrivent dans la formation. Chaque métropole (Grand Reims et Troyes Métropole) disposera d'une plateforme locale créant le lien avec l'ensemble des plateformes régionales et portant également l'articulation avec l'ouverture internationale.

Dans le cadre de ce projet, je suis en charge d'une partie de l'investissement pour mettre en lien les différents acteurs de l'ancienne région Champagne-Ardenne autour de l'axe « Robotique et Production (PROBOT) » (1 500 k€). Plusieurs partenariats sont d'ores et déjà mis en place.

Afin de disposer d'une vitrine technologique à destination des industriels, une première plateforme « Industrie du Futur » sera mis en place avec le pôle formation UIMM Champagne-Ardenne. Cette plateforme, en lien avec Platinium 3D (plateforme technologique et scientifique dédiée à l'obtention de pièces métalliques par les procédés de fabrication additive), aura pour objectif d'illustrer différents concepts de l'industrie du futur (flexibilité, personnalisation, cobotique …) au travers l'assemblage d'un vrai produit (Figure 3-16).

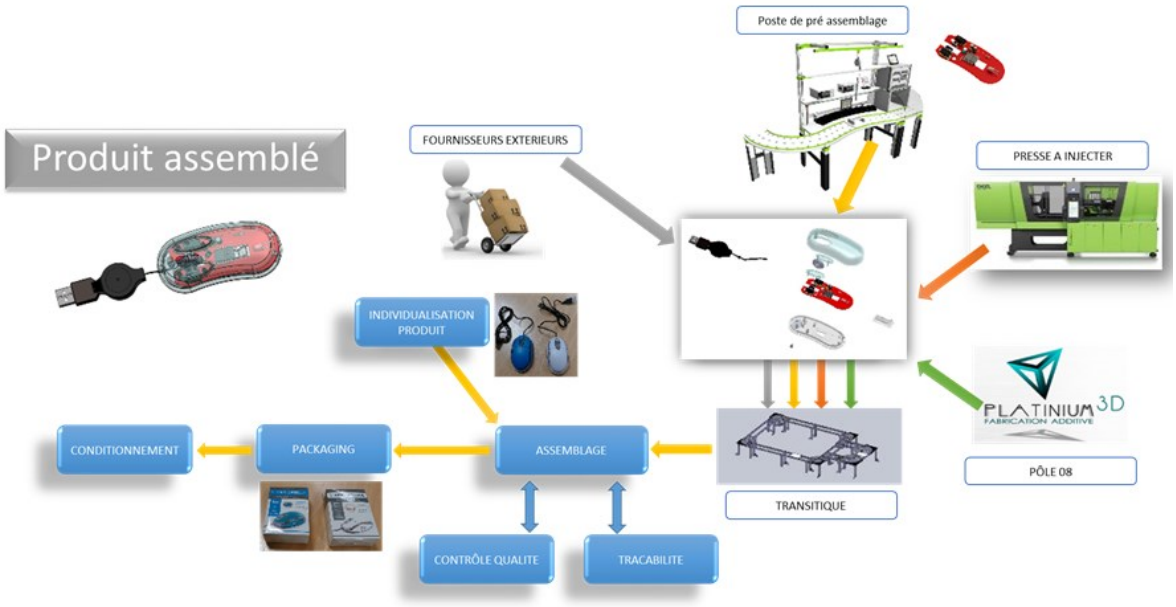

Figure 3-16 : Projet de plateforme à destination des industriels

Une seconde plateforme, cette fois-ci côté recherche, devrait permettre l'interconnexion avec différentes équipes de recherche (URCA et UTT) et des plateformes technologiques

existantes (Centre-Image, Roméo, Cellflex4.0) (Figure 3- 17). Ainsi, une salle de *virtual commissioning* industriel avec captation vidéo est en cours d'installation au laboratoire CReSTIC.

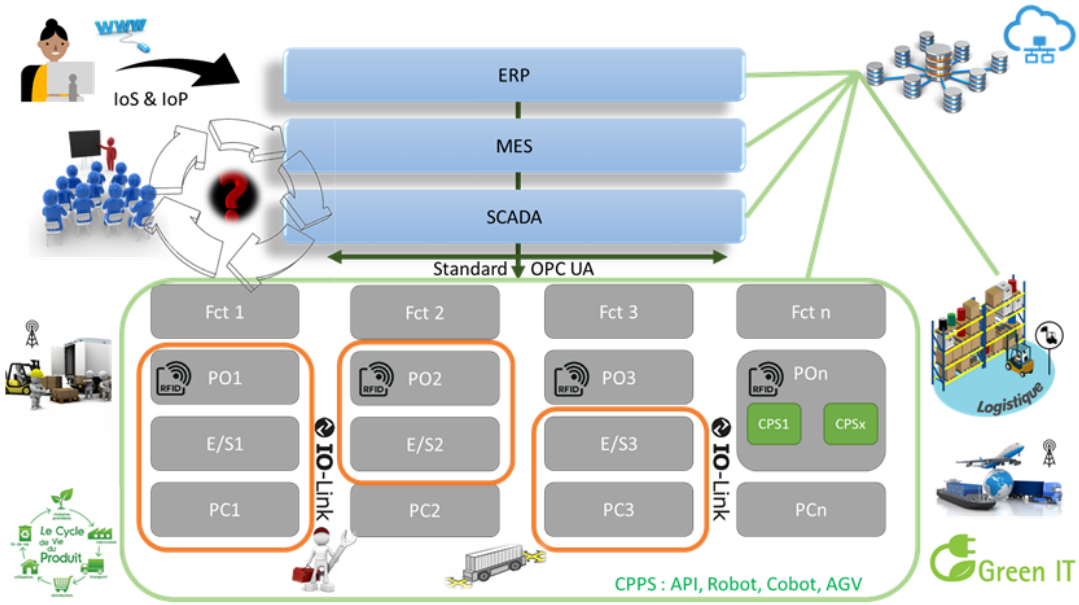

Figure 3- 17 : Projet de plateforme à destination des chercheurs

Le projet FFCA sera un atout majeur pour la mise en œuvre de nos différentes activités de recherche et permettra de donner une visibilité à nos travaux en direction des industriels pour conclure des partenariats industriels.

CONCLUSION GENERALE

Ce mémoire d'Habilitation à Diriger des Recherches a permis de synthétiser mes différentes activités depuis ma nomination en tant que Maître de Conférences. J'ai ainsi repris les trois missions (Enseignement, Recherche et Responsabilité) de l'Enseignant-Chercheur dans les 2 premiers chapitres.

Le chapitre 1 est un CV développé montrant mon activité en pédagogie, mes participations et responsabilités prises depuis 2007 que ce soit localement ou nationalement. Il permet également de reprendre les encadrements doctoraux qui sont ensuite détaillés au chapitre 2.

L'objectif du deuxième chapitre n'était pas de détailler les différents travaux de Recherche, mais de synthétiser ceux-ci en y montrant les évolutions au cours du temps. On y retrouve notamment les deux grandes thématiques de ma recherche : la commande et le diagnostic à base de modèles des Systèmes à Evénements Discrets. Concernant les approches de commande, j'y reprends l'approche historique du laboratoire autour d'une méthodologie de synthèse à base de SCT, et y développe les évolutions de celles-ci autour de structures décentralisées/distribuées mais également des approches par contraintes logiques. Pour le second thème, le diagnostic a été un nouveau thème pour le laboratoire lors de mon arrivée en doctorat. J'ai pu ensuite collaborer autour de cette thématique avec de nouveaux collègues et/ou au travers de projets inter-établissements. Cette section constitue un lien logique avec mon projet de recherche présenté en chapitre 3.

Les travaux réalisés en partenariat industriels ont permis de mettre l'accent sur la place de l'Opérateur Humain au sein du système. En continuité avec mes travaux en cours, mon envie personnelle, mais aussi avec mes récentes responsabilités au sein de l'URCA m'ont conduit à proposer au chapitre 3 un projet de recherche centré sur la place de l'Opérateur Humain dans l'Industrie du Futur. Je souhaite en effet montrer que les outils et/ou modèles utilisés dans les TIC (Technologies de l'Information et de la Communication) ne sont pas là uniquement pour

répondre à la problématique du « Pour quoi ? », mais aussi du « Pour qui ? ». Dans ce sens, des recherches doivent être menées pour :

- accompagner le concepteur dans son initiative à trouver un ensemble de contraintes pouvant être appliqué à la commande par filtre,
- aider, rassurer, l'utilisateur dans sa prise de décision dans les approches de diagnostic en prenant en compte les concepts de *Situation Awareness* et/ou *Dissonance*.

Par ailleurs, un lien fort existe entre la commande et le diagnostic des systèmes. Les travaux récemment entamés sur la commande tolérante aux fautes et sur la reconfiguration des Systèmes Automatisés de Production méritent d'être poursuivis. Les réflexions de l'équipe CDSED sur ces sujets doivent bien entendu mûrir en prenant en compte l'environnement scientifique, les différents travaux déjà réalisés, mais aussi les partenariats académiques et industriels à envisager.

Au-delà de l'aspect scientifique de ces perspectives de recherche, il convient également de faire évoluer nos formations. J'ai déjà pu introduire une initiation à la recherche sur certains modules, mais au titre du trio « Savoir, Savoir-faire, Savoir-être », nous nous devons de modifier nos méthodes de travail en pédagogie pour nos futurs automaticiens 4.0.

Enfin, je conclurai personnellement sur cette réflexion qui me fait dire que, étant a priori un garçon **discret**, j'ai bien choisi mon domaine. Curieux, j'essaie d'être à l'écoute des autres et d'observer mon environnement, comme pourrait l'être un module de **diagnostic**. J'aspire également au travers cette habilitation à prendre plus de responsabilités, à prendre donc la **commande**. Toutefois, ne croyez surtout pas que je suis un **modèle**, car je ne suis qu'un **homme**.

Bibliographie

Alanche, P, Lhoste, P., Morel, G., Roesh, M., Salim, M., Salvi, P. (1986). Application de la modélisation de la Partie opérative à la structuration de la commande, Journée AFCET, Montpellier, 1986.

Baddeley, A. Human memory: theory and practice. Hove: Lawrence Erlbaum Associates, 1990.

Baier C., Katoen J. and Larsen K. Principles of model checking. MIT press, 2008.

Balemi S., Hoffmann G.J., Gyugyi P., Wong-Toi H., Franklin G.F., "Supervisory control of a rapid thermal multiprocessor", Proceeding IEEE Transactions on Automatic Control, Vol. 38, N°7, pp.1040-1059, 1993.

Benlorhfar, R., Annebicque, D., Gellot F., Riera B. (2011). Robust filtering of PLC program for automated systems of production, 18th World Congress of the International Federation of Automatic Control, Milan, Italie, August 2011.

Bérard B., Bidoit, M., Finkel, A., Laroussinie, F., Petit, A., Petrucci, L., Schnoebelen, P. Systems and Software Verification: Model-Checking Techniques and Tools. Springer, 1999.

Bévan R., Berruet P. and de Lamotte F. Generation of multiplatform control for transitic systems using a component-based approach. Proceedings of IEEE 17th International Conference on Emerging Technologies & Factory Automation (ETFA 2012), 2012.

Cai K. and Wonham M. Supervisor Localization: A Top-Down Approach to Distributed Control of Discrete-Event Systems. IEEE Transactions on Automatic Control, 55(3):605{618, mars 2010. ISSN 0018-9286

Carré-Ménétrier V. and Zaytoon J. "Grafcet: behavioural issues and control synthesis", European Journal of Control (EJC), vol 8 n°4, 2002.

Cassandras C.G., Lafortune S., "Introduction to Discrete Event Systems", Kluwer Academic Publisher, ISBN 978-0-387-68612-7, 2008.

Chaillet-Subias A. Approche multi-modèles pour la commande et la surveillance en temps réel des systèmes à évènements discrets. Thèse de Doctorat, Toulouse 3, 1995.

Chapurlat V. and Bonjour E. From Model Based Systems Engineering to Model Based System Realization: role and relevance of IVTV Plan, APMS (1) 2014:

Cnockaert, J.C. and Floru, R. (1991). Introduction à la psychophysiologie du travail. Nancy: PUN, 1991.

Combacau M., “Commande et surveillance des systèmes à événements discrets complexes : application aux ateliers flexibles”, Thèse, Toulouse, 1991.

Coupat R. «Méthodologie pour les études d’automatisation et la génération automatique de programmes Automates Programmables Industriels sûrs de fonctionnement : Application aux Equipements d’Alimentation des Lignes Electrifiées». Thèse de doct. Université de Reims Champagne-Ardenne, 2014.

Debernard, S., Riera, B., Poulain, T. (2013). ‘L'espace de travail commun pour l'aide à la supervision et à la coopération homme-machine’. In P. Millot, Ergonomie des systèmes homme-machine : Conception et coopération (Traité IC2, série Systèmes automatisés), Hermès, pp. 295-341, ISBN 9782746239203, 2013.

Debernard, S., Riera, B., Poulain, T. (2014). Chapter 8 : The Common Work Space for the Support of Supervision and Human–Machine Cooperation, Designing Human-machine Cooperation Systems, Patrick Millot, Wiley InterScience, ISBN 978-1-84821-685-3, pp 285-342, 2014.

Debouk R., Lafortune S., Teneketzis D., “Coordinated decentralized protocols for failure diagnosis of discrete event systems”, Discrete Event Dynamic Systems: Theory and Applications, Vol. 10, N°1-2, pp.33-86, January 2000.

Delorme X., Dolgui A., Kovalev S. and Kovalyov M. Minimizing the number of workers in a paced mixed-model assembly line. European Journal of Operational Research 272(1): 188-194, 2019.

Dimanche V. «Compréhension fine du comportement des lignes des réseaux RER, métro et tramway pour la réalisation des études d’exploitabilités». Thèse de doct. Université de Reims Champagne-Ardenne, 2018.

Du D., Gu J. and Pardalos P.M. Satisability Problem: Theory and Applications: DIMACS Workshop, March 11-13, 1996. American Mathematical Soc., January 1997. ISBN 978-0-8218-7080-8.

Faure J.M. and Lesage J.J. Methods for safe control systems design and implementation. 10th IFAC Symposium on Information Control Problems in Manufacturing, INCOM'2001, Sep. 2001.

Fri M., Belmajdoub F. and Lefebvre D. Fault diagnosis of discrete event systems under time constraints. 2nd International Conference on Systems Informatics, Modelling and Simulation (SIMS), 2016

Frizon de Lamotte F. Proposition d’une approche haut niveau pour la conception, l’analyse et l’implantation des systèmes reconfigurables. Thèse de Doctorat de l’Université de Bretagne Sud, 2006.

Hietter, Y. Synthèse algébrique de lois de commande pour les systèmes à évènements discrets logiques, thèse de doctorat, Cachan, École normale supérieure, 2009.

Hill R., Tilbury D. and Lafortune S. «Modular supervisory control with equivalence-based

conflict resolution». In: American Control Conference, 2008.

Holloway L.E., Krogh B.H., “Fault detection and diagnosis in manufacturing systems”, International Conference on Computer Integrated Manufacturing, Vol. 2, pp.252-259, 1990.

Hu H., Chen C., Su R, Liu Y. and Zhou M. «Distributed supervisor synthesis for automated manufacturing systems using Petri nets». In: IEEE International Conference on Robotics and Automation (ICRA), 2014.

International Electrotechnical Commission, “PLCs – Part 3: programming languages”, Publication 611131-3, 1993.

International Electrotechnical Commission, “GRAFCET specification language for sequential function charts”, IEC 60848, 2002.

International Electrotechnical Commission, Functional Safety of Electrical/Electronic/Programmable Electronic Safety-related Systems, IEC61508.

El Khattabi S., “Intégration de la surveillance de bas niveau dans la conception des systèmes à événements discret”, Thèse de Doctorat, Lille, France, 1993.

Komenda J. and Lin F. «Modular supervisory control of networked discrete-event systems». In: 13th International Workshop on Discrete Event Systems (WODES), 2016.

Kumar R., “Supervisory Synthesis Techniques for Discrete Event Dynamical Systems”, Thesis for Ph. D. Degree, University of Texas, 1991.

Lefebvre D. On-Line Fault Diagnosis With Partially Observed Petri Nets. IEEE Transactions on Automatic Control, Vol. 59, N°7, July 2014.

Leplat, J. (1997). Regard sur l’activité en situation de travail. Paris: PUF, 1997.

Lhoste P., “Surveillance des M.S.A.P.: les Atouts de la modélisation de Comportement”, Journée Surveillance du pôle SED (GT2) du GR Automatique, Paris, 1991.

Lin F. and Wonham M. «Decentralized supervisory control of discrete-event systems». In: Information sciences 44.3,1988.

Marangé P., Gellot F. and Riera B. Remote control of automation systems dor training. IFAC/IFIP/IFORS/IEA Symposium, Analysis, Design, and Evaluation of Human Machine Systems (ADEHMS'07), Sep 2007, Séoul, South Korea. pp.CD, 2007.

Marangé, P. (2008). Synthèse et filtrage robuste de la commande pour des systèmes manufacturiers sûrs de fonctionnement. Thèse de doctorat, Université de Reims Champagne-Ardenne, 2008.

Marangé P., Philippot A., Pétin J.F. and Gellot F. Diagnosability evaluation by model-checking. 9th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes (SAFEPROCESS'15), Paris, France, September 2015.

Millot P., Ergonomie des systèmes homme-machine : Conception et coopération. Hermés, Paris, France. 2013.

Niang M., Philippot A., Gellot F., Coupat R., Riera B. and Lefebvre S. Formal Verification for Validation of PSEEL's PLC Program. 14th International Conference On Informatics in Control, Automation and Robotic (ICINCO'17), Madrid, Spain, July 2017.

Niang M., Vérification formelle et simulation pour la validation des programmes API des EALE. Thèse de doctorat, Université de Reims Champagne-Ardenne, 2018

Parasuraman, R., Sheridan, T., and Wickens, C. (2000). A model for types and levels of human interaction with automation. IEEE Transactions on Systems, Man and Cybernetics, SMC-30(3), 286–297, 2000.

Philippot A. «Contribution au diagnostic décentralisé des systèmes à événements discrets : Application aux systèmes manufacturiers».Thèse de doct. Université de Reims Champagne-Ardenne, 2006.

Philippot A. and Tajer A. From GRAFCET to Equivalent Graph for Synthesis Control of Discrete Events Systems. 18th Mediterranean Conference on Control and Automation (MED10), pp 683-688, IEEE, Marrakech, Morocco, June 2010.

Pichard R., Philippot A., Saddem R and B. Riera. Safety of Manufacturing Systems Controllers by Logical Constraints With Safety Filter. IEEE Transactions on Control Systems Technology (TCST), Issue 99, Vol. PP. 2018. 10.1109/TCST.2018.2827329.

Pichard R., Contribution à la Commande des Systèmes à Événements Discrets par Filtre Logique. Thèse de doctorat, Université de Reims Champagne-Ardenne, 2018

Qamsane Y., Tajer A. and Philippot A. «Synthesis and implementation of distributed control for a flexible manufacturing system ». In: Second World Conference on Complex Systems (WCCS), 2014.

Qamsane Y., Tajer A. and Philippot A. «A synthesis approach to distributed supervisory control design for manufacturing systems with GRAFCET implementation». In: International Journal of Production Research 55.15, 2017.

Qamsane Y., Tajer A. and Philippot A. «Towards an approach of synthesis, validation and implementation of distributed control for AMS by using events ordering relations». In: International Journal of Production Research 55.21, 2017.

Ramadge P.J.G., Wonham W., "Supervisory control of a class of discrete events systems", SIAM journal of Control and Optimization, Vol. 25, N°1, 1989.

Ramadge P.J.G., Wonham M., "The Control of Discrete Event Systems", Proceeding IEEE, Vol. 77, N°1, January 1989.

Richard, J.-F. Faut-il revoir la notion de charge mentale ? Psychologie Française, 41 (4), 309-312, 1996.

Riera, B., Debernard, S. (2003). Basic Cognitive Principles Applied to the Design of Advanced Supervisory Systems for process control, Handbook of Cognitive Task Design, Erick Holnagel (Ed.), Lawrence Erlbaum Associates, pp 255-281, 2003.

Riera, B. (2001). Contribution à la conception d'outils de supervision adaptés à l'homme. Habilitation à Diriger des Recherches, Université de Valenciennes et du Hainaut-Cambrésis, décembre 2001.

Riera B., Philippot A., Coupat R., Gellot F. and Annebicque D. (2015a). A non-intrusive method to make safe existing PLC Program. 9th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes (SAFEPROCESS'15), Paris, France, September 2015.

Riera B., Philippot A., Annebicque D. et Gellot F. (2015b). La commande par contraintes logiques de sécurité : principe, applications et mise en œuvre. 10ème Colloque sur la Modélisation des Systèmes Réactifs (MSR 2015), Nancy, France, novembre 2015.

Roussel J.M., "Analyse de GRAFCETs par Génération Logique de l'Automate Equivalent", Thèse de l'Ecole Normale Supérieure de Cachan, 1994.

Roussel J.M. and Denis B. Safety properties verification of ladder diagram programs. Journal Européen des Systèmes Automatisés (JESA), Lavoisier, 2002.

Rudie K. and Wonham M. «Think globally, act locally: Decentralized supervisory control». In: IEEE transactions on automatic control 37.11, 1992.

Saddem R., Toguyeni A. K. A., and Tagina M., "Algorithme d'interprétation d'une base de signatures temporelles causales pour le diagnostic en ligne des Systèmes à événements Discrets," 9th International Conference of Modeling, Optimization and SIMulation, MOSIM 2012. Bordeaux, France, 2012.

Saddem R. and Philippot A. Utilisation de diagnostiqueurs discrets pour l'établissement de Signatures Temporelles Causales pour le diagnostic des SED. Colloque International sur le Monitoring des Systèmes Industriels 2014 (CIMSI'2014), Marrakech Maroc, December 2014.

Sampath M., Sungupta R., Lafortune S., Sinnamohideen K., Teneketzis D., "Diagnosability of discrete event systems", In 11th International Conference Analysis Optimization of Systems: Discrete Event Systems, Sophia-Antipolis, France,1994.

Sampath M., "A Discrete Event Systems Approach to Failure Diagnosis", Thesis, University of Michigan, 1995.

Sargent, L. D. and Terry, D. J. (2000). The moderating role of social support in Karasek's job strain model. Work and Stress, Vol. 14, No. 3, pp. 245-261, 2000.

Schafaschek G., De Queiroz M. and Cury J. «Local modular supervisory control of timed discrete-event systems». In: IEEE Transactions on Automatic Control 62.2 (2017).

Stanton, N. A., Young, M., and McCaulder, B. (1997). Drive-by-wire: The case of driver workload and reclaiming control with adaptive cruise control. Safety Science, 27(2/3), 1997.

Su R., “Distributed Diagnosis for Discrete-Event System”, Thesis of PhD, University of Toronto, Canada, 2004.

Tajer A., Philippot A., Gellot F., Carré-Ménétrier V., “Contribution à l’amélioration de la praticabilité des approches formelles de synthèse de commande”, Journées Doctorales d’Automatique 2003, Valenciennes, pp.239-244, 25-27 juin 2003.

Tajer A., Philippot A. and Carré-Ménétrier V. Decentralized Implementation Approach of Control Synthesis of Manufacturing Systems. 2nd International Conference on Multimedia Computing and Systems (ICMCS'11), IEEE, Ouarzazate, Morocco, April 2011.

Vanderhaegen F. and Caulier P. A multi-viewpoint system to support abductive reasoning. Information Sciences, Vol. 18, Issue 24, December 2011.

Vanderhaegen F. Dissonance Engineering: A New Challenge to Analyse Risky Knowledge When using a System. International Journal of Computers Communications & Control. ISSN 1841-9836, 9(6):776-785, December, 2014.

Vanderhaegen F. and Carsten, O. Can dissonance engineering improve risk analysis of human–machine systems? Cognition, Technology & Work, 19 (1). pp. 1-12. ISSN 1435-5558, 2017.

Wickens, C. D., Gordon, S. E., Liu, Y. An introduction to human factors engineering, 1998.

Wonham M. Supervisory Control of Discrete-Event Systems. In Encyclopedia of Systems and Control, pages 1396-1404. Springer, London, 2015.

Young, M. S., Stanton, N. A. Mental workload: theory, measurement, and application. In W. Karwowski (Ed.), International encyclopedia of ergonomics and human factors: Vol. 1, pp.507-509. London: Taylor and Francis, 2001.

Zaytoon J. and Carré-Ménétrier V., Synthesis of a correct control implementation for manufacturing systems, International Journal of Production Research, vol. 39, n°2, p. 329-345, 2001.

Zaytoon J. and Lafortune S. Overview of fault diagnosis methods for Discrete Event Systems. Annual Reviews in Control, 37, 2013

Zaytoon J. and Riera B. Synthesis and implementation of logic controllers – A review. Annual Reviews in Control, March 2017. ISSN 13675788. URL http://linkinghub. elsevier.com/retrieve/pii/S1367578816301043.

Zhong H. and Wonham M. «On the consistency of hierarchical supervision in discrete-event systems» .In: IEEE Transactions on automatic Control 35.10 (1990).

ANNEXES

***R. Pichard*, A. Philippot, R; Saddem and B. Riera**. Safety of Manufacturing Systems Controllers by Logical Constraints With Safety Filter. IEEE Transactions on Control Systems Technology (TCST), Issue 99, Vol. PP. 2018. 10.1109/TCST.2018.2827329 (JCR - 2016 Impact Factor: 3.882), (Q1 en 2017 dans Scimago Journal & Country Rank)

***Y. Qamsane*, A. Tajer and A. Philippot**. A synthesis approach to distributed supervisory control design for manufacturing systems with GRAFCET implementation. International Journal of Production Research (IJPR), Issue 15, Vol. 55. 2017. http://dx.doi.org/10.1080/00207543.2016.1235804 (JCR - 2016 Impact Factor: 2.325), (Q1 en 2017 dans Scimago Journal & Country Rank)

Safety of Manufacturing Systems Controllers by Logical Constraints with Safety Filter

Romain PICHARD*, Alexandre PHILIPPOT, Ramla SADDEM and Bernard RIERA

***Abstract*—This paper presents an approach to safe controller synthesis for manufacturing systems controlled by Programmable Logic Controllers (PLC). In this work, manufacturing systems are considered as Discrete-Event Dynamic Systems (DEDS) with logical inputs and outputs. The methodology is based on the use of safety constraints placed at the end of the PLC program. These constraints are checked off-line by a formal approach and acted as a safety filter in order to be robust against control errors. The proposed approach separates the functional control part from the safety part and focuses on the latter. This paper presents the whole methodology and recent improvements on consistency checking of a set of Boolean expressions.**

***Keywords*—Discrete-Event Dynamic Systems, Logical Filter, Manufacturing Systems, Programmable Logic Controllers, Safe Control.**

I. Introduction

In parallel with the complexity increase of the automated production systems in term of quantity, requirements in communication, diversity of the components, etc., users' requirements concerning the dependability and the design assistance of control programs also increase. Indeed, the Programmable Logic Controller (PLC) constitutes one of the main architectures of manufacturing system control and is programmed with standardized languages (IEC 61131-3). Automatic control engineer does not have until now any actual help for the control program design. In fact, designing safe control programs for manufacturing systems is mainly based on human competencies and experiences. Hence, to ensure, in a formal way the safety of the automated production systems is a real scientific challenge carrying out valuable industrial issues.

In this work, manufacturing systems are considered as Discrete-Event Dynamic Systems (DEDS) [1] with logical inputs and outputs [2], controlled by Programmable Logic Controller (PLC), programmed with standardized languages [3]. In this paper we will only consider controller as being a PLC program.

[4] describes a framework to synthesize a supervisor for DEDS: Supervisory Control Theory (SCT). SCT is based on automaton models (language theory) of plants (physical systems under control) and specifications. From these models, a supervisor is computed automatically. But in practice, this leads to a state explosion that restricts the applicability of SCT to small systems [5], [6]. Moreover, SCT is based on the asynchronous hypothesis (one event at each time), but in a PLC several variables may change simultaneously due to the cycle time, so the asynchronous hypothesis is not guaranteed. Finally, SCT deals only with 'supervisor synthesis' (i.e. prohibit event), but in practice, we also need a 'controller synthesis' approach (i.e. force event).

Against these problems, some approaches are based on Boolean algebra instead of automaton models and language theory. Algebraic synthesis introduces the formalism, the theorems, and the algorithms to solve a Boolean equations system [7]. Authors propose a formal approach to compute PLC control program from requirements given in natural language. [8] proposes an approach (safety filter) in order to guarantee the safety regardless of the control program already in the PLC. This safety filter is based on a set of Boolean equations which define safety constraints. For a given set of input variables and the first affectation of output variables, the safety filter, must find the nearest affectation to the first affectation which does not violate any safety constraint. It is worth to note that the safety filter is implemented online.

This paper presents for the first time the entire approach of safety filter design with the latest updated formalism and the algorithm. Compared to previous work [8], the consistency of the set of safety constraints is now defined, improvements on the formalism allow to use more complicated constraints and the algorithm has been changed in order to take into account the previous improvements.

The main idea is to separate the safety part from the functional part. The functional part can be expressed with common approaches like automata or Petri net. Regarding the safety part, it is defined as a set of Boolean expressions called Safety Constraints (*CS*). For a given system, the safety constraints are formally defined from a structural and dysfunctional analysis of the system. These safety constraints ensure the safety of the production system and its products in case of errors in the control program. The proposed method has the risk of interfering the necessary operations to complete the manufacturing process or even blocking the whole process, but it ensures the safety. At the moment, this work assumes there are no failures on the plant. The methodology integrates different off-line verifications of these constraints, which allow to validate the safety constraints and to check the safety filter robustness. The safety constraints are expressed as a product of logical variables: Inputs/Outputs (*I*/*O*) and possible internal variables. The safety filter is then implemented at the end of the PLC control program in order to ensure the safety no matter the control program already implemented in the PLC.

This paper is organized as follows. Section II is a review of control synthesis approaches which have oriented this research work. An overview and the formalism of the safety filter approach is presented in Section III. Then, a two steps approach to analyze the set of constraints is detailed in Section IV. Finally, a discussion around this work is proposed with some perspectives.

II. State of the art

Given a manufacturing system (plant), we are interested in the synthesis of a controller which follows a set of specifications. In the first instance, the general approach can be described as a closed-loop between the plant and the controller (Fig. 1). The outputs of the plant are sensors values and considered by the controller as uncontrollable events (Σ_u). The inputs of the plant are actuators orders and considered by the controller as controllable events (Σ_c). So the role of the controller is to compute the value of each output in order to follow the specifications.

Several methods have been proposed to synthesize a Discrete-Events Dynamic Systems (DEDS) control law [9]. Before describing the proposed safety filter approach, some approaches from the literature are briefly presented.

Manuscript received April 12, 2018;

The authors are with the Research Centre for Science and Information Technology and Communication, University of Reims Champagne Ardennes, Reims, France (e-mail: romain.pichard@univ-reims.fr; alexandre.philippot@univ-reims.fr; ramla.saddem@univ-reims.fr; bernard.riera@univ-reims.fr).

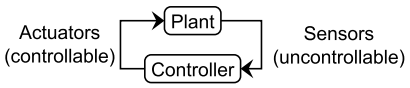

Figure 1. Controlled system

A. Supervisory Control Theory

Supervisory Control Theory (SCT) [4] is based on the distinction between models of the plant (system), properties to respect and structure of control [10]. The current approach of using this theory is based on three steps. The first step is to build two models: the model of the plant (representing the possible behaviors of the system), and the model of specifications (expected properties). From these two models, the next step is to obtain the mathematical model of the plant under control. The final step is to synthesize a supervisor that will restrict the possible behaviors of the plant. So the behavior of the supervisor is then the same as the plant under control. The advantage of the SCT is, in addition to the formal framework proposed, the separation between the models which allows us to avoid mixing several concepts into one model (plant models and specification models). The disadvantage of the SCT is the difficulty to apply it in practice because of modeling problems and combinatory explosion due to the mathematical tools used (language theory). Indeed that restricts the applicability of SCT to small systems [5], [6].

As real control applications involve the use of large size systems, the main issue hindering the use of SCT in the industry is the state-space explosion. To deal with this problem, several approaches have been proposed in the literature. The modular supervisory control consists of designing a set of small supervisors to meet various individual specifications, rather than constructing a single monolithic supervisor that simultaneously meets all the specifications [11]. The decentralized approach suggests dividing the global supervision's objective into several sub-objectives [12]. The resulting individual sub-supervisors are simultaneously executed to implement a solution for the initial problem. The hierarchical control structure is based on the use of simplified process models to develop high-level supervisors capable to take overarching decisions [13]. These decisions are transmitted to the low-level supervisors that control the real process. Multi-level hierarchical and organizational control generalized the hierarchical control [14], [15]. The compositional approach is based on the construction of a set of small automata equivalent to the monolithic representation of the problem. Using this set, the final supervisor is computed with a smaller state space exploration than the classical synthesis approach [16]. Symbolic approach proposes to use BDD structure in order to compute efficiently a supervisor, this supervisor is defined by logic conditions [17].

Most of these approaches are based on a language theory modeling (for plant and specification), and the associated algorithms can imply a **combinatory explosion**. In an industrial context there are several problems with these approaches, firstly the plant model may be difficult to obtain due to the complexity of the system. Secondly, usually there are a lot of sensors and actuators, so the used algorithm can take too much time before providing a solution. In addition, for each modification or correction on the models, the entire algorithms have to be computed. Finally, the synchronous behavior of the PLC can involve simultaneous changes of the input or output vectors. This synchronous behavior is not easy to take into account in the SCT approach (which is by definition **asynchronous**). Some extensions exist to adapt the SCT to PLC programming approach, [18]–[22]. These works propose several ways to design implementable controller in a PLC. Even if these solutions brought a PLC point of view or tried to abolish the asynchronous hypothesis, that is not entirely satisfactory in an industrial context. Indeed the formalism and the tools used for SCT are not well known by industrial practitioners.

We propose to detail two approaches which introduce the main concepts of our approach: separating of the functional part from safety part and Boolean algebra for DEDS control design.

B. Supervised Control Concept

The SCT distinguishes the plant and the specifications. But for a manufacturing system, the specification model can be difficult to obtain. Indeed the model must include two tasks: control and supervision. [5] describes a framework to separate control and supervision tasks. In their *Supervised Control Concept*, the plant (system to be controlled) is coupled with a controller (Fig. 2). The plant is seen as a DEDS which produces uncontrollable events (Σ_u), the controller is also a DEDS that controls the plant by producing controllable events (Σ_c). So the *controller* is able to *force* some controllable events. For an external observer, the coupled plant and controller (called "extended plant") is seen as a device that evolves spontaneously. So the *supervisor* is confined to *prohibit* some events.

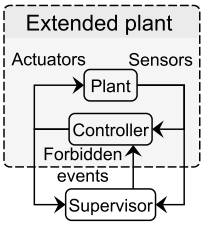

Figure 2. Supervised control concept

The advantage of this framework is the conciseness of the final models due to the separation between the control and supervisory tasks. The disadvantage is the dependency between the controller and the supervisor. Indeed, the controller must be known to design the supervisor. So the supervisor must be changed if the controller is changed.

C. Algebraic Synthesis

Due to the difficulty to model and synthesize a controller, [7] proposes an algebraic method of control synthesis. Mathematical foundations of algebraic synthesis are based on classical Boolean algebra. So each specification can be expressed by a Boolean formula without any restrictions on forms. Moreover it is possible to specify if a variable is *Known* or *Unknown* (sensors and actuators for DEDS). The goal of algebraic synthesis is to find a parametric Boolean formula for each *Unknown* variable, solutions are expressed by a composition of parameters and *Known* variables. The choice of a particular solution is then left to the expert by choosing the value of parameters. This approach is very interesting from a theoretical point of view and the synthesis law obtained is easily implementable in a PLC. However, it is difficult to take into account the different operating modes. Indeed there is no separation between functional and safety part, moreover, the program safety is not guaranteed because the sufficiency of the constraints set is not formally checked. Therefore a change in specifications necessarily entails a new formulation of requirements, so a new calculation of the entire solution.

D. The proposed approach

Based on the previous works and discussions, we propose an approach of safe control which incorporates some principles of these previous works (Fig. 3).

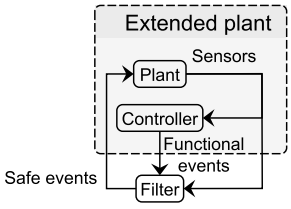

Figure 3. Safety filter approach

The principle of models' separation from [5] is conserved and extended. Indeed, in order to simplify the design of PLC controllers in an industrial context, we need to distinguish functional control part and safety control part [23]. In the proposed approach, the functional control part expresses *what the system must do* (i.e how to achieve the production goals) and is placed in the *Controller*. The safety control part expresses *what the system must never do* and is placed in a *Filter*.

In [4], [5] and all the SCT-based approach, a *controllable* event is defined as an event which can be forbidden. The proposed safety filter approach is a control approach and not a supervisor approach, in this context, a *controllable* event is defined as an event which can be forced. So, the controller (functional part) and the safety filter (safety part) can force events.

The mathematical foundations are based on some of the theoretical results proved in algebraic synthesis works [24]. However, there are several fundamental differences with the algebraic synthesis method proposed in [7]. Indeed requirements (i.e. specifications) are only relating to safety and not to the functional part. Moreover, the implementation is not the same in order to simplify the usage of the safety filter for automation engineers in practice.

The safety filter is designed to allow a permissive control (large control space allowed). The safety filter is valid regardless of the control law and therefore whatever the functional specifications. The separation between the functional part and safety part in our approach allows a much easier management of operating modes. Indeed a change of functional requirements does not imply a change of the safety filter. Finally, there is no asynchronous hypothesis due to the formalism and the algorithm used, but the assumption of no failure on the plant is conserved.

Finally, the global methodology to design a safety filter is as follow. Firstly, the safety requirements are defined by an expert of the system using risks analysis. Then, a set of safety constraints are used to translate the safety from requirements to specifications. After, these safety constraints have to be checked to ensure their consistency (no contradiction) and their sufficiency (able to avoid all dangerous situations). At last the safety filter can be implemented into the PLC cycle in order to check the functional controller and changes values of actuators (controllable variables) if necessary.

III. The safety filter formalism

This section details the formalism linked to the safety filter. Firstly, a general definition of the safety filter is provided. Then the safety constraints and associated functions are defined. At last, priorities between controllable variables into a constraint are presented.

The notations used in the following section are based on the Boolean algebra and PLC programming. 0 means FALSE and 1 means TRUE. $x \in \{0,1\}$ is a Boolean variable. "." "+" "$\overline{x}$" are respectively the logical operators AND, OR and NOT. $\sum$ and $\prod$ are respectively the logical sum (OR) and the logical product (AND) of logical variables. t is the current scan time (from PLC point of view), $t-1$ the previous PLC scan time. O is the set of output variables (actuators: controllable variables) and there are N_o outputs. I is the set of input variables (sensors: uncontrollable variables) and there are N_i inputs. $x[t]$ is the value of the variable x at the t^{th} PLC scan time (for simplicity, $x[t]$ will be noted x). $x_k[t]$ is the logical variable corresponding to the k^{th} variables at the t^{th} PLC scan time (for simplicity, $x_k[t]$ will be noted x_k).

In this work it is assumed that the PLC scan time is sufficient to detect any change of the input vector (**synchronous** operations, possible simultaneous changes of PLC inputs' states). In addition, the plant is considered functioning normally **without failure**.

In this paper, we propose to define the safety filter as follow (Def. 1).

Definition 1. *A **safety filter** is a 4-uple $\langle Y, O, CSs, CSc \rangle$ such that:*

- *Y set of uncontrollable variables (sensors, internal variables).*
- *O set of controllable variables (actuators).*
- *CSs set of Simple Safety Constraints (see in Def. 2).*
- *CSc set of Combined Safety Constraints (see in Def. 3).*

Each part of this definition is discussed and defined in this section.

A. Constraints formalism

In the proposed approach, the safety requirements are translated into a set of Boolean equations called safety constraints. These safety constraints are defined as logical product functions. A safety constraint is composed by uncontrollable variables (Y) and controllable variables (O). Two kinds of safety are proposed: *Simple* Safety Constraints (Def. 2) and *Combined* Safety Constraints (Def. 3).

In this paper the logical product of a set of Boolean variable A is defined as $\prod(A)$ (Eq. (1)).

$$\prod(A) = \prod_{k=1}^{card(A)} (x_k), x_k \in A \tag{1}$$

Definition 2. *A **Simple Safety Constraint** (CSs) is defined as a product of uncontrollable variables and only 1 controllable variable.*

Several constraints can exist for a specific output variable o_k. The number of *CSs* for o_k is depict $Ncss^k$. Considering $i \in [1, Ncss^k]$, $k \in [1, N_o]$, $Y0_i^k \subseteq Y$ and $Y1_i^k \subseteq Y$, then the i^{th} *CSs* of o_k is defined Eq. (2).

$$CSs_i^k = \begin{cases} \prod(Y0_i^k).o_k \\ OR \\ \prod(Y1_i^k).\overline{o_k} \end{cases} \tag{2}$$

$Ncss_1^k$ is the number of *CSs* with o_k and $Ncss_2^k$ is the number of *CSs* with $\overline{o_k}$. So, it exists $Ncss^k = Ncss_1^k + Ncss_2^k$ *CSs* for the controllable variable o_k.

Combining each of these constraints leads to define two polynomial functions $F0s^k$ and $F1s^k$ (sum of products, $\sum\prod$). These functions are only composed by uncontrollable variables associated to o_k (Eq. (3)).

$$\begin{array}{rcl} \sum_{i=1}^{Ncss^k} CSs_i^k & = & \sum_{i=1}^{Ncss^k} (\prod(Y0_i^k).o_k + \prod(Y1_i^k).\overline{o_k}) \\ & = & \sum_{i=1}^{Ncss_1^k} (\prod(Y0_i^k)).o_k + \sum_{i=1}^{Ncss_2^k} (\prod(Y1_i^k)).\overline{o_k} \\ \sum_{i=1}^{Ncss^k} CSs_i^k & = & F0s^k.o_k + F1s^k.\overline{o_k} \end{array} \tag{3}$$

These functions will be used later in this paper in order to model the set and reset functions for the controllable variable o_k.

Definition 3. *A **Combined Safety Constraint** (CSc) is defined as a product of uncontrollable variables and several controllable variables.*

The number of *CSc* is $Ncsc$. Considering $j \in [1, Ncsc]$, $Y_j \subseteq Y$ and $O_j \subseteq O$, then the j^{th} *CSc* is defined Eq. (4).

$$CSc_j = \prod(Y_j) \ . \ \prod(O_j) \tag{4}$$

B. Meaning and usage of safety constraints

In the safety filter approach, a constraint (*CSs* or *CSc*) is said *violated* when it is equal to *true*, and a constraint is said *solved* when it is equal to *false*. So a safety constraint means "if the uncontrollable part is true, then the controllable part must be false".

The aim of the safety filter is to guarantee that all the values of controllable variables are valid regarding the safety requirements. Therefore, we define a safety condition (Def. 4), which will be used by the safety filter to check the set of constraints.

Definition 4 (Safety condition)**.** *A set of constraints is solved if the controllable variables ($N_o = dim(O)$) solve all the constraints.*

$$\sum_{k=1}^{N_o} \sum_{i=1}^{Ncss^k} CSs_i^k + \sum_{j=1}^{N_{CSc}} CSc_j = 0 \tag{5}$$

Considering this safety condition and the Eq. (3), the functions $F0s^k$ and $F1s^k$ indicate if the controllable variable o_k must be forced to 0 or 1. Indeed, if $F0s^k = 1$, o_k must be forced to 0 in order to solve the constraint CSs_i^k (Eq. (3)). In other words, $F0s^k$ is the *simple reset function* of o_k and $F1s^k$ is the *simple set function* of o_k.

C. Constraint resolving

A violated constraint must be solved online in order to guarantee the corresponding safety requirements. To solve a violated constraint, the values of controllable variables must be changed. It exists two kinds of constraints (*CSs* and *CSc*) and then the way to solve them is different.

1) CSs resolving: The only way to solve a *CSs* is to complement the value of its controllable variable. Considering a CSs_i^k violated, if the uncontrollable part is $true$ so the controllable part must be $false$ to solve it. The resolving of a *CSs* is then structurally deterministic.

2) CSc resolving: If a Combined Safety Constraint (*CSc*) is violated, a choice must be made to solve it depending on the numbers of controllable variables. Indeed, for a Combined Safety Constraint with 2 controllable variables, three possibilities exits to solve it (forced one or both of the controllable variables). Consequently, for n controllable variables, a solution must be chosen from the $2^n + ... + 2^0 = 2^{n+1} - 1$ possibilities. The resolving of a *CSc* is then not deterministic. In order to make deterministic resolution of *CSc*, a notion of priority between controllable variables must be included. The problem of solving a set of safety constraints (Simple and Combined) is discussed in Section IV-C.

D. Priority definition

Given a specific affectation of uncontrollable variables which violates at least one *CSc*, the way to resolve constraints online must be always the same in order to be deterministic. Offline, we propose to associate to each *CSc* a *priority*. This priority is a choice made by the designer and allows to indicate which output has priority against others in this *CSc*.

In this paper, we propose to introduce and define 2 levels of priorities. The first one is given by the structure of each combined constraint, and the second one is given by the choice made during the design of constraints.

a) Structural priority: For a *CSc* with 2 controllable variables, if one of them is forced by a *CSs* the corresponding variable is temporarily uncontrollable for this *CSc*. In this condition, the *CSc* may be seen as a *CSs* because there is only 1 controllable variable free. So there is only 1 way to resolve the constraint: complement the value of the last controllable variable.

This structural priority can be extend to *CSc* with more than 2 controllable variables. Let a be an uncontrollable variable and $\{o_1, o_2, o_3\}$ 3 controllable variables, the structural rule for the constraint $CSc_1 = a.o_1.o_2.\overline{o_3}$ is:

$$a.(o_1.F1s^2.F0s^3 + F1s^1.o_2.F0s^3 + F1s^1.F1s^2.\overline{o_3}) = 0$$

With this structural priority, if only 1 controllable variable o_k is free, due to the activation of *set* or *reset* simple functions, the *CSc* is seen as a *CSs*. These rules are automatically computed according to the structure of *CSc* and used to synthesize the safety filter (Section IV-C).

b) Chosen priority: When at least 2 controllable variables are being free in a combined constraint, there exist several ways to solve this *CSc*. In this condition, the expert has to define which controllable variables have to be forced in order to solve the constraint.

This choice is specified in the constraint definition by the expert. Then it is used to check and synthesize the safety filter (Section IV). As the structural priority, it is possible to define by a Boolean formula the chosen priority.

According to the translation of priorities into Boolean equations, it is possible to define set and reset functions for each actuator o_k based on *CSc* instead of *CSs* previously. These functions are called $F0c^k$ for combined reset function and $F1c^k$ for combined set function. They are defined by combining structural and chosen Boolean equations of priorities for each controllable variables.

Based on the constraints definition and the priorities chosen by the expert, the simple and combined set and reset functions are computed automatically. A complete example is provided Section V, constraints and functions are listed.

IV. Analysis and synthesis of safety filter

In order to guarantee the safety and to validate the safety filter, the problem (constraints and priorities) must be analyzed off-line. We propose two steps of verification, the first step is a consistency checking of the problem (contradiction between constraints and/or priorities). The second one is a sufficiency verification in order to check if the filter is able to guarantee the safety of any controller.

A. Consistency checking

In this paper, we propose to define the *consistency* of our problem (constraints and priorities), and a graph analysis approach to prove it. This part is the main improvement compared to previous works, indeed we propose a necessary and sufficient condition instead of only some necessary conditions.

Definition 5. *Considering a set of uncontrollable variables (Y), a set of controllable variables (O), a set of safety constraints (CS) and a set of priorities (P): the problem (constraints and priorities) is consistent if and only if, whatever the uncontrollable variables, there always exists at least one interpretation of O that satisfies all the safety constraints and priorities.*

Our problem may sound like SATisfiability (SAT) problem in computer science [25]. Indeed given a formula composed of Boolean variables, a SAT solver must replace consistently each variable by TRUE or FALSE in order to evaluate the formula to TRUE. In this case, the formula is called satisfiable. But our problem is different:

- There are two kinds of variable: uncontrollable and controllable variables;
- We must prove there exists a solution for all the possible values of the uncontrollable variables.

We propose in this paper a two steps approach to ensure the consistency. Firstly a structural analysis of constraints without priorities is done. This step allows us to reduce the initial problem. Then, based on the reduced problem and the priorities, a reachability analysis

is proposed to provide a necessary and sufficient condition for the consistency of the reduced problem and consequently for the initial problem (Fig. 4).

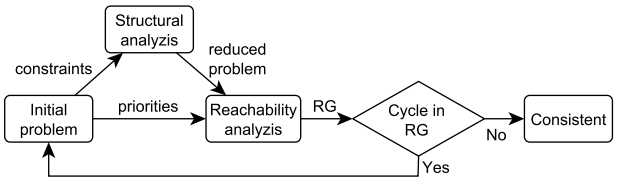

Figure 4. Method to check the consistency

1) Structural analysis: In order to reduce the number of constraints for the consistency checking, we represent the structural interaction between constraints with an undirected graph called Structural Graph SG = (CS, E) such that:

- $CS = CSs \cup CSc$ is a set of vertices: each vertex is a safety constraint;
- E is a set of edges. Each edge represents a link between 2 constraints.

An edge between 2 constraints means that if one of them is violated, the resolution may violate the other.

Definition 6. *Given 2 constraints CS_i and CS_j, an edge exists between them if:*

1- *Logical product of uncontrollable part of CS_i and CS_j is not always false;*
AND
2- *If $O_k \in CS_i$ then $\overline{O_k} \in CS_j$ OR If $\overline{O_k} \in CS_i$ then $O_k \in CS_j$*

In other words, an edge can exist between 2 constraints only if they can be violated at the same time no matter the controllable variables (first point in Def. 6). Moreover, if there are no shared (and complementary) controllable variables between constraints, edges cannot exist (second point in Def. 6).

In the structural graph, some constraints may be isolated (no edges with other constraints) so regarding the consistency problem, these constraints should be ignored. Indeed the resolution of these constraints will never violate other constraints, and other constraints will never violate them. Based on this reduced problem, a reachability analysis is proposed below to check the consistency.

2) Reachability analysis: The goal is to construct a graph which represents all possible developments of the variables during the solving procedure. Each vertex of the graph is defined by an affectation of variables and labeled by constraints which are violated. Each arc represents the action made to solve the constraints. We define a Reachable Graph RG = (V, A) as a directed graph such that:

- V is a set of vertices: each vertex is a full affectation of the uncontrollable (Y) and controllable variables (O);
- A is a set of arcs which represents the action made by the solving procedure.

Given a reduced problem (Section IV-A1), RG is built as follows:

1- Compute each safe vertex;
2- Compute each unsafe vertex;
3- Link vertices according to the priorities;
4- Delete safe vertices which are not linked.

Computation of safe vertices is possible with a SATisfiability problem solver (SAT) [25]. Given a set of clauses (Boolean equations) a SAT solver is able to provide all possible affectations of variables which satisfy all the clauses.

Each affectation which is not listed by SAT solver is obviously unsafe. For each of them, we have to analyze the affectation in order to know which constraints are violated. We labeled each unsafe vertex with the corresponding violated constraints.

For each unsafe vertex and considering the violated constraints we apply the priorities. As shown previously the resolution is deterministic for an affectation, so there exists only one outgoing arc from each vertex. If for an affectation, two constraints are in contradiction (one force to $false$ and another to $true$ a variable) the arc is changed to a loop on the same unsafe vertex.

Finally, the safe vertices without any entering or outgoing arcs are removed in order to reduce the size of the final graph.

3) Consistency condition: Each vertex of the reachability graph represents an affectation of variables at the beginning of the safety filter algorithm (Alg. 1). So each vertex may be initiated and given an initial vertex, the final value of actuators will be always the same because the resolution procedure (through priorities) is deterministic.

Based on the definition and construction of RG, the consistency condition (Section IV-A) can be translated to a reachable analysis (Proposition 1).

Proposition 1. *The problem is consistent if and only if for all unsafe vertices, a safe vertex is reachable.*

Checking each path of the graph is time-consuming. But regarding the construction of RG, there is only 1 outgoing arc from each vertex. So the consistency condition can be summarized to the non-existence of a cycle in RG (Proposition 2). Indeed if a cycle exists, it means that there exists at least 1 unsafe affectation which cannot be changed to a safe affectation.

Proposition 2. *The problem is consistent if and only if a cycle does not exist in the reachability graph.*

With this analysis, it is possible to check the consistency of the problem (constraints and priorities). So there is no logical or structural contradiction in the definition of constraints and priorities. However, the aim of the safety filter is to avoid all the dangerous situations. This property is called sufficiency and has to be ensured formally off-line. It can be noted that the consistency is necessary but not sufficient to ensure the sufficiency of the safety filter.

B. Sufficiency checking

A solution to ensure the sufficiency is to use a model-checker to validate the constraints set [26]. These works are summarized in this paper but not detailed. The main idea is to model with timed and communicating automata the behavior of the system's components, the product, and the most permissive controller. The most permissive controller means that there are no hypotheses about the controller and all the possible outputs changes are considered.

An expert makes a dysfunctional analysis to define all the dangerous states, which must never be reached by the elements of the plant or products. Then he identifies for each dangerous state the subset of the plant (system components and products flow sequence in interaction) to model for the validation by model-checking. All these dangerous states are marked in the model-checking by temporal logic or manually. So the validation of the safety filter consists in verifying that we can not reach a marked state in the model-checker.

With this approach we are able to check some properties:

- **Sufficiency**: no matter the path of the plant controlled by the most permissive controller and safety filter, it never reaches any dangerous state.
- **Reachability**: we can check if some states are reachable, these states are the objectives described in the functional requirements. It is a proof for the existence of a controller.

After having chosen and verified the constraints set, the safety filter definition is over. So whether the affectation of actuators values sent by the functional controller, the final values sent to the system is safe. The last point is now to implement the filter into a PLC.

C. Safety filter synthesis

1) Solving procedure: From now, the set of constraints is considered to be consistent. The resolving of a set of constraints is not trivial because the solving of one can imply the violation of another constraint and so on. To avoid this, and be able to have a safe solution, we propose a procedure to achieve it. This procedure is illustrated in Fig. 5 and is presented below.

The plant generates some uncontrollable events which are given to the functional controller. The latter computes the expected values of actuators based on the functional requirements. Then the safety filter checks the safety and changes the actuators values if necessary. Finally, the safe outputs values are sent to the plant.

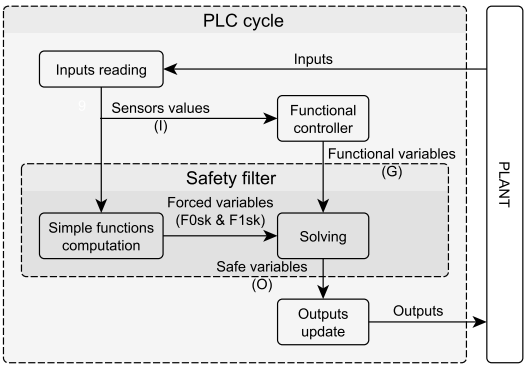

Figure 5. PLC cycle with safety filter

For the safety filter synthesis, we must introduce g_k as a functional variable of the set G given by the functional controller, and mem_k as a temporary variable to update the output variable o_k.

The safety filter solves the set of constraints as follow. Firstly, the simple safety constraints are solved by forcing some controllable variables. Then the combined safety constraints are solved according to the forced variables and the priorities by forcing other controllable variables. As the problem is consistent, a solution is always available. Next, if a controllable variable remains free, the value of the functional outputs ($g_k \in G$) is applied to the corresponding controllable variable o_k. At the end, the final value of each controllable variable o_k is used to update the outputs signals.

2) Solving algorithm: The proposed algorithm follows the solving procedure describes before. It can be easily implemented in a PLC by using the Structured Text (ST) language. The algorithm uses the *set* and *reset* functions described earlier in this paper. In this section, we propose to detail how they are used online to compute a safe solution.

At each PLC cycle, there are two inputs set: Y is a set of uncontrollable variables (sensors, internal variables, ...) and G given by the functional controller. Both are considered as constant and uncontrollable by the safety filter in a cycle.

a) Simple functions computation: The algorithms begins by computing simple set/reset functions $F1s^k$ and $F0s^k$.

b) CSs resolution: Theorem 1 describes the resolution and consistency condition for a single-unknown equation (Cf. Theorem 11 in [24] for proof).

Theorem 1. *Let $\mathbb{B}$ be a Boolean algebra, let a, b, $x \in \mathbb{B}$. The solution of the equation $a.\bar{x}+b.x=0$ is $x=a+\overline{b}.p$ with $p \in \mathbb{B}$ a parameter iff $a.b=0$.*

The Theorem 1 is used to solve all the CSs_i^k of controllable variable o_k. Indeed for each output o_k, the equation $o_k \,.\, F0s^k + \overline{o_k} \,.\, F1s^k = 0$ needs to be verified (see Eq. (3) and (5)). The solution is $o_k = \overline{F0s^k} \,.\, p + F1s^k$ with $p \in \mathbb{B}$ a parameter. In our case this parameter is the functional value g_k. This computation is used to solve the *CSs* and to initialize the temporary variables mem_k (Eq. (6)).

$$mem_k = \overline{F0s^k} \,.\, g_k + F1s^k \tag{6}$$

mem_k corresponds to the temporary values of controllable variables o_k. If the set/reset simple functions are $false$, then $mem_k = g_k$. Else, mem_k is forced.

c) CSc resolution: Based on the values of mem_k initialized before, the *CSc* are updated. If at least 1 *CSc* is violated by current vector *mem*, a $Flag$ is activated. In this condition the set/reset combined functions ($F1c^k$ and $F0c^k$) are updated. Then, the new values of mem_k are computed using simple and combined set/reset functions, this computation is also based on Theorem 1 (Eq. (7)).

$$mem_k = \overline{F0s^k + F0c^k} \,.\, g_k + (F1s^k + F0c^k) \tag{7}$$

Since Eq. (7) contains also simple functions, the simple constraints will not be violated during the resolution of combined constraints.

d) Outputs update: At last, the final values of the mem_k which solve all the constraints are sent to the plant by updating the outputs values of the PLC (O).

```
Data: No
Input: G, Y
Output: O
/* F0s^k and F1s^k calculation
   (Eq. (3))                    */
for k = 1, ..., No do
    F0s(k) := ...;
    F1s(k) := ...;
end
/* Simple constraints
   resolution and
   initialization               */
for k = 1, ..., No do
    mem(k) := NOT F0s(k) AND G(k)
    OR F1s(k);
end
Flag := true;
/* Combined resolution          */
while Flag do
    CSC := cscUpdate(mem,Y);
    Flag := cscIsViolated(CSC);
    if Flag then
        for k = 1, ..., No do
            F0c(k) := ...;
            F1c(k) := ...;
        end
        for k = 1, ..., No do
            mem(k) := NOT(F0s(k)
            OR F0c(k)) AND G(k)
            OR (F1s(k) OR F1c(k));
        end
    end
end
/* Output update                */
for k = 1, ..., No do
    O(k) := mem(k);
end
```

Algorithm 1: Safety filter algorithm

V. Illustrative systems

This section illustrates all points presented in the paper by an example. The example is described below. Let a and b be two uncontrollable variables, O_1, O_2 and O_3 three controllable variables and the constraints presented Table I.

Table I. Initial problem

CSs_1	=	$\overline{a}.b.O_1$	CSs_2	=	$a.O_2$
CSc_1	=	$\mathbf{O_1}.\overline{O2}$	CSc_2	=	$O_2.\overline{\mathbf{O_3}}$

A. Chosen priority

Variables in **bold** in a *CSc* have priority against others controllable variables in this constraint. For example, CSc_1 violated ($=1$) means that $O_1 = 1$ and $O_2 = 0$. To solve it logically it exists 3 ways:

1- priority to O_1: O_2 must be forced to true ($O_2 = 1$)
2- priority to O_2: O_1 must be forced to false ($O_1 = 0$)
3- no priority: O_1 and O_2 must be forced ($O_1 = 0$ and $O_2 = 1$)

Based on the Table I, the chosen priority for CSc_1 correspond to the first bullet.

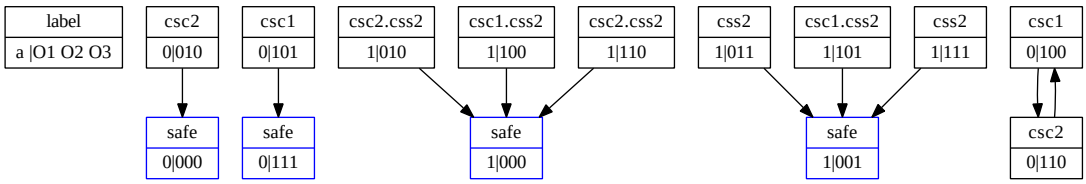

Figure 6. Reachability Graph for the initial problem (Table I)

B. Set/reset functions

Based on *CSs*, simple set/reset functions are computed automatically for each controllable variable (Table II).

Table II. SIMPLE SET/RESET FUNCTIONS

O_k	Set ($F1s^k$)	Reset ($F0s^k$)
O_1	0	$\overline{a}.b$
O_2	0	a
O_2	0	0

Based on *CSc* and chosen priorities, combined set/rest functions are computed (Table III). The first parenthesis correspond to the structural priority, the second one (if exists) to the chosen priority.

Table III. COMBINED SET/RESET FUNCTIONS

O_k	Set ($F1c^k$)	Reset ($F0c^k$)
O_1	(0)	($F0s^2$)
O_2	($F1s^1$)+ ($F1s^1.\overline{F0s^2}.mem_1.\overline{mem_2}$)	($F0s^3$)+ ($F1s^2.\overline{F0s^3}.mem_2.\overline{mem_3}$)
O_3	($F1s^2$)	(0)

C. Consistency checking

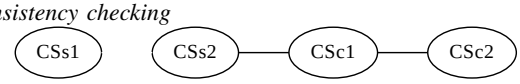

Figure 7. Structural graph for example in Table I

At this point, the set of constraints must be analyzed. We propose to present only the consistency checking for this example due to constraints on space.

The first step is to reduce the problem by removing isolated constraints. This analyze is based on the Structural Graph (SG). This graph is constructed only by using the constraints and structural priorities.

On Fig. 7, there is no edge between CSs_1 and CSs_2 because $(\overline{a}.b).(a) = 0\ \forall(a,b)$ (first point in def 6). Moreover there is no edge between CSs_1 and CSc_1 due to second point in def 6. At last there are no shared controllable variables between CSs_1 and CSc_2 so no edge between them. The isolated constraints CSs_1 in Fig. 7 must be removed in order to obtain the reduced problem.

From the reduced problem and the chosen priority (Section III-D), the resulting Reachability Graph RG is presented in Fig. 6. For this example and with these chosen priorities, the problem is not consistent. Indeed, from vertex $a|O_1O_2O_3$ = 0|100 or 0|110 it is not possible to reach a safe vertex by applying the priority.

The cycle is between CSc_1 and CSc_2. We have to analyze these constraints and the chosen priorities. Indeed, from the affectation $a = 0$, $O_1 = 1$, $O_2 = 1$ and $O_3 = 0$, the variable O_2 will be alternatively forced to *false* then *true* indefinitely. A solution to solve this inconsistency is to change the chosen priority. The new set of constraints to test is the same as previously (Table I) but the priority for CSc_1 is now: $CSc_1 = O_1.\overline{\mathbf{O2}}$.

The same consistency analysis is computed, the Structural Graph is unchanged because only the priorities changes and the new Reachability Graph is provided Fig. 8. No cycle exists in this graph, so the problem is consistent.

Finally, the safety filter algorithm can be used online to check and ensure the safety requirements. More complex and manufacturing examples are described in [8] (palletizer), [27] (sorting boxes).

VI. DISCUSSIONS

The proposed approach of the safety filter is based on Boolean equations to specify the safety. There is no risk of combinatory explosion during the synthesis of the safety filter. Furthermore, the formalism and algorithm are designed to take into account synchronous occurrences of events. Then the approach allows a full separation of the functional part from safety part, so the work habits of engineer can remain unchanged. Lastly, the proposed algorithm can be easily implemented in a PLC with standardized ST language.

Resolving constraints to guarantee the safety can be in opposition to the notion of "productivity" or "liveness". Indeed, modified the expected behavior of a system by a correction of an output can, in the worst case, lead to a deadlock situation. The controllability of a system must be studied and can be a criterion for the choice of the chosen priority defined in the Section III-D.

The main difficulty of this proposition can be identified by the capacity of the expert to establish the set of constraints. It can be seen as a complex exercise but some helping "rules" are possible. Indeed, constraints concerning a classical component (as a cylinder or a

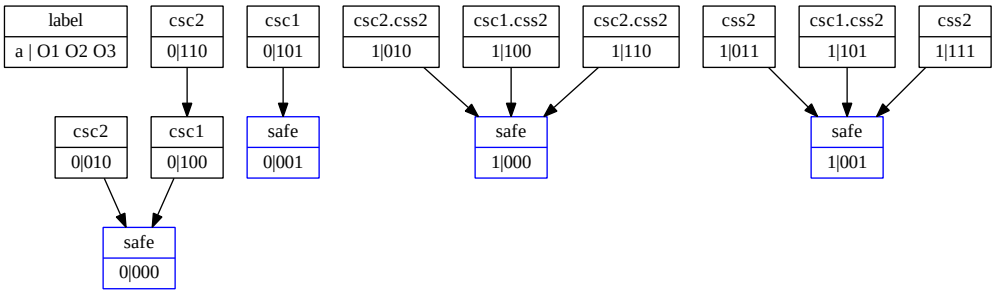

Figure 8. Reachability Graph for the consistent problem

conveyor) can be quickly identified into a library. The expert can then focus on expressing constraints of interactions between components or between component and product. The flow of the product is a guiding element. Some works are in progress to obtain automatically this set of constraints.

Another way to synthesize the proposed safety filter is to use an online SAT solver. A proof of concept of this proposition is proposed in [27]. By using optimization criteria during the computation, this approach allows to not use the priority for solving combined constraints. However, to the authors' knowledge, there is no SAT solver designed for PLC. Authors are currently trying to propose a dedicated SAT solver for the safety filter.

VII. Conclusion

This paper has proposed a method based on the use of a safety filter to avoid control errors (represented as a set of logical constraints) and make an existing PLC program safe. The safety filter is based on a set of Boolean expressions which define safety constraints (from safety requirements). The final orders sent to the plant are safe whatever the controller. The proposed approach allowed to fully separate functional and safety parts in order to simplify the whole controller design. Moreover, the capability of the filter to guarantee the safety is formally checked off-line. An algorithm (easily written in normalized ST language) has been proposed to implement the safety filter in any PLC. At last, we have shown through an example how the constraints are designed and the advantages to separate the safety from functional.

The prospects of this work identify a range of work areas:

- There is no specific initial state, so it is well adapted to manage different **running modes** (manual mode, run-stop, ...).
- Applying this approach on **flexible** or **re-configurable** manufacturing systems could be really useful due to the separation of functional and safety.
- Due to the structure of constraints ("If ... Then ..."), the application to job and activity **scheduling** is studied.
- Currently, when a constraint is violated, we used it to detect a control error, but this information could also be used to detect or **diagnose** a fault of the plant.

References

[1] C. G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*. Springer Science & Business Media, Dec. 2009.

[2] S. Balemi, G. Hoffmann, P. Gyugyi, H. Wong-Toi, and G. Franklin, "Supervisory control of a rapid thermal multiprocessor," *IEEE Transactions on Automatic Control*, vol. 38, no. 7, pp. 1040–1059, Jul. 1993.

[3] P. IEC 61131-3, "Programming languages for programmable logic controllers," 2013.

[4] P. Ramadge and W. Wonham, "The control of discrete event systems," *Proceedings of the IEEE*, vol. 77, no. 1, pp. 81–98, Jan. 1989.

[5] F. Charbonnier, H. Alla, and R. David, "Discrete-event dynamic systems," *Control Systems Technology, IEEE Transactions on*, vol. 7, no. 2, pp. 175–187, 1999.

[6] J. N. Vilela and P. N. Pena, "Supervisor abstraction to deal with planning problems in manufacturing systems," in *2016 13th International Workshop on Discrete Event Systems (WODES)*, 2016, pp. 117–122.

[7] Y. Hietter, J.-M. Roussel, and J.-J. Lesage, "Algebraic synthesis of transition conditions of a state model," in *9th International Workshop on Discrete Event Systems, 2008. WODES 2008*, May 2008, pp. 187–192.

[8] B. Riera, R. Coupat, A. Philippot, F. Gellot, and D. Annebicque, "Control design pattern based on safety logical constraints for manufacturing systems: application to a palletizer," in *Discrete Event Systems*, vol. 12, Cachan, France, May 2014, pp. 388–393.

[9] J. Zaytoon and B. Riera, "Synthesis and implementation of logic controllers – A review," *Annual Reviews in Control*, Mar. 2017. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1367578816301043

[10] G. Faraut, L. Pietrac, and E. Niel, "Formal Approach to Multimodal Control Design: Application to Mode Switching," *IEEE Transactions on Industrial Informatics*, vol. 5, no. 4, pp. 443–453, Nov. 2009.

[11] D. Gouyon, J.-F. Pétin, and A. Gouin, "A pragmatic approach for modular control synthesis and implementation," *International Journal of Production Research*, vol. 2, no. 14, pp. 2839–2858, 2004. [Online]. Available: https://hal.archives-ouvertes.fr/hal-00120780

[12] L. Feng and W. M. Wonham, "Supervisory Control Architecture for Discrete-Event Systems," *IEEE Transactions on Automatic Control*, vol. 53, no. 6, pp. 1449–1461, Jul. 2008.

[13] K. C. Wong and W. M. Wonham, "Hierarchical control of discrete-event systems," *Discrete Event Dynamic Systems*, vol. 6, no. 3, pp. 241–273, Jul. 1996. [Online]. Available: https://link.springer.com/article/10.1007/BF01797154

[14] R. C. Hill, J. E. R. Cury, M. H. de Queiroz, D. M. Tilbury, and S. Lafortune, "Multi-level hierarchical interface-based supervisory control," *Automatica*, vol. 46, no. 7, pp. 1152–1164, Jul. 2010. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0005109810001597

[15] K. T. Seow, "Organizational Control of Discrete-Event Systems: A Hierarchical Multiworld Supervisor Design," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 1, pp. 23–33, Jan. 2014.

[16] S. Mohajerani, R. Malik, and M. Fabian, "A Framework for Compositional Synthesis of Modular Nonblocking Supervisors," *IEEE Transactions on Automatic Control*, vol. 59, no. 1, pp. 150–162, Jan. 2014.

[17] Z. Fei, S. Miremadi, K. Åkesson, and B. Lennartson, "Efficient Symbolic Supervisor Synthesis for Extended Finite Automata," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 6, pp. 2368–2375, Nov. 2014.

[18] M. Fabian and A. Hellgren, "PLC-based implementation of supervisory control for discrete event systems," in *Proceedings of the 37th IEEE Conference on Decision and Control, 1998*, vol. 3, 1998, pp. 3305–3310 vol.3.

[19] A. Vieira, J. Cury, and M. de Queiroz, "A Model for PLC Implementation of Supervisory Control of Discrete Event Systems," in *IEEE Conference on Emerging Technologies and Factory Automation, 2006. ETFA '06*, Sep. 2006, pp. 225–232.

[20] F. Basile and P. Chiacchio, "On the Implementation of Supervised Control of Discrete Event Systems," *IEEE Transactions on Control Systems Technology*, vol. 15, no. 4, pp. 725–739, Jul. 2007.

[21] M. Moreira and J. Basilio, "Bridging the Gap Between Design and Implementation of Discrete-Event Controllers," *IEEE Transactions on Automation Science and Engineering*, vol. 11, no. 1, pp. 48–65, Jan. 2014.

[22] A. D. Vieira, E. A. P. Santos, M. H. d. Queiroz, A. B. Leal, A. D. d. P. Neto, and J. E. R. Cury, "A Method for PLC Implementation of Supervisory Control of Discrete Event Systems," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 1, pp. 175–191, Jan. 2017.

[23] R. Cuer, L. Piétrac, E. Niel, S. Diallo, N. Minoiu-Enache, and C. Dang-Van-Nhan, "A formal framework for the safe design of the Autonomous Driving supervision," *Reliability Engineering & System Safety*, vol. 174, pp. 29–40, Jun. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0951832017305914

[24] J.-M. Roussel and J.-J. Lesage, "Design of Logic Controllers Thanks to Symbolic Computation of Simultaneously Asserted Boolean Equations," *Mathematical Problems in Engineering*, vol. 2014, p. Article ID 726246, May 2014, 15 pages.

[25] D. Du, J. Gu, and P. M. Pardalos, *Satisfiability Problem: Theory and Applications : DIMACS Workshop, March 11-13, 1996*. American Mathematical Soc., Jan. 1997.

[26] P. Marangé, F. Gellot, and B. Riera, "Industrial risk prevention by robust filter for manufacturing control system," in *Fault Detection, Supervision and Safety of Technical Processes*, Sants Hotel, Spain, Jun. 2009, pp. 1348–1353.

[27] R. Pichard, N. Ben Rabah, V. Carre-Menetrier, and B. Riera, "CSP solver for Safe PLC Controller: Application to manufacturing systems," *IFAC-PapersOnLine*, vol. 49, no. 12, pp. 402–407, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2405896316309119

International Journal of Production Research

ISSN: 0020-7543 (Print) 1366-588X (Online) Journal homepage: http://www.tandfonline.com/loi/tprs20

A synthesis approach to distributed supervisory control design for manufacturing systems with Grafcet implementation

Yassine Qamsane, Abdelouahed Tajer & Alexandre Philippot

To cite this article: Yassine Qamsane, Abdelouahed Tajer & Alexandre Philippot (2017) A synthesis approach to distributed supervisory control design for manufacturing systems with Grafcet implementation, International Journal of Production Research, 55:15, 4283-4303, DOI: 10.1080/00207543.2016.1235804

To link to this article: http://dx.doi.org/10.1080/00207543.2016.1235804

International Journal of Production Research, 2017
Vol. 55, No. 15, 4283–4303, https://doi.org/10.1080/00207543.2016.1235804

A synthesis approach to distributed supervisory control design for manufacturing systems with Grafcet implementation

Yassine Qamsane[a]*, Abdelouahed Tajer[a] and Alexandre Philippot[b]

[a]*Laboratory of Electrical Engineering and Control Systems, National School of Applied Sciences, Cadi Ayyad University, Marrakech, Morocco;* [b]*Research Centre for Science and Information Technology and Communication, University of Reims Champagne Ardennes, Reims, France*

(*Received 21 December 2015; accepted 6 September 2016*)

In supervisory control, computational complexity and implementation flexibility represent major challenges when a large number of local components compose a target system. To overcome these difficulties, we propose a formal approach to distributed control synthesis and implementation for automated manufacturing systems (AMS). We assume that the system is modelled with automata in a local modular fashion. Local control specifications are defined for each local subsystem by means of logical equations to construct local controllers (LCs). Then, global control specifications, stated as logical implications, are applied to the LCs, which allows synchronisation and cooperative interaction among the subsystems. This paper makes two contributions. First, it outlines a formal method for constructing minimally restrictive and deadlock-free distributed controllers (DCs). Second, it proposes a method for the interpretation of these DCs into Grafcet, which is a graphical modelling formalism widely used to design the controller's dynamic behaviour for AMS. An experimental manufacturing system illustrates the approach.

Keywords: discrete-event dynamic systems; manufacturing systems; supervisory control; synthesis methods; Grafcet; PLC

1. Introduction

The development of computer and information technologies in recent decades has led to the rise of discrete event dynamic systems (DEDS) that become an inherent part of the world. A wide variety of physical systems deriving from modern technology such as communication protocols, automated manufacturing systems (AMS), air traffic systems, control systems in automobiles, logistic systems, are viewed as DEDS. DEDS are characterised by asynchronous occurrences of discrete events (Cassandras and Lafortune 2008). This work is interested in AMS, which are a class of DEDS used to produce products or to perform services. Different techniques and methods may help the practitioner in the analysis, design, validation, implementation, control and optimisation of AMS, such as the verification and validation (V&V) method and the synthesis method. The V&V method consists of checking that an AMS model meets given specifications, and that it achieves its intended purpose (Baier and Katoen 2008; Biallas, Brauer, and Kowalewski 2012). A major drawback of this method is the need for prior writing of the programmes. For example, in the case of a redesign, a programmer cannot fully validate a programme when a change is carried out on expected properties. The synthesis method (see, e.g. Ramadge and Wonham 1987; Komenda, Masopust, and van Schuppen 2012; Cury et al. 2015) is based on constructing models of the system and their expected properties in order to obtain a control model, which in essence, meets the specified properties. The latter are therefore taken into account from the beginning of the design, which enables swift reaction to blockings and changes. Among synthesis methods, supervisory control theory (SCT) initiated by Ramadge and Wonham (1987) has considerably enhanced results in the DEDS domain. It provides formal control architectures based on properties such as controllability, observability, safety, liveness, and ultimately, diagnosability. The SCT is basically supported by automata and formal languages (Hopcroft, Motwani, and Ullman 2006). Its objective is to define (synthesise) a supervisor which disables the occurrence of a set of events in such a way that the supervised DEDS behaves in accordance with the considered specifications.

As real control applications involve the use of large size systems, two main issues hinder the use of SCT in the industry: the state-space explosion and the models interpretation, i.e. supposing the computation is successful, the understanding of large models remains challenging. To deal with these problems, several approaches have been proposed in

*Corresponding author. Email: qamsaneyassine@gmail.com

the literature. The modular supervisory control (see, e.g. Wonham and Ramadge 1988; Gouyon, Petin, and Gouin 2004; Silva et al. 2011) consists of designing a set of small supervisors to meet various individual specifications, rather than constructing a single monolithic supervisor that simultaneously meets all the specifications. The decentralised approach (see, e.g. Feng and Wonham 2008; Sayed-Mouchaweh, Philippot, and Carre-Menetrier 2008; Shu and Lin 2014) suggests to divide the global supervision's objective into several sub-objectives. The resulting individual sub-supervisors are simultaneously executed to implement a solution for the initial problem. The hierarchical control structure (see, e.g. Wong and Wonham 1996; Hill et al. 2010; Seow 2014) is based on the use of simplified process models to develop high-level supervisors capable to take overarching decisions. The latter are transmitted to the low-level supervisors that control the real process.

The work presented in this paper investigates a distributed control structure. Distributed approaches assume that a system is composed of several interconnected subsystems. To reach a global goal, the subsystems are required to exchange data with each other. The main idea is to design local controllers (LCs) which control each subsystem individually, and request the other controllers to share enough information to cooperatively execute their control actions. A number of distributed architectures have been developed and deployed in the literature for AMS. The method presented in (Wang, Mahulea, and Silva 2013) is a distributed model predictive control approach for systems modelled as timed continuous Petri nets (PNs). It assumes that the subsystems are connected by places modelling buffers. Each subsystem is controlled by a LC which can access all the local variables as well as its buffer places. In this approach, the subsystems convergence from their given initial states to their desired final ones is addressed algorithmically without use of a high-level coordinator. However, even though the approach applies to timed models, it doesn't consider the minimum-time state evolution issues. The method proposed in (Hu et al. 2014, 2015; Yang, Hu, and Liu 2015) aims to synthesise distributed liveness enforcing supervisors to control large-scale AMS in the framework of PN. Although, this approach delivers relevant theoretical insights to reduce computational complexity, its application to complex systems is still lacking.

In this paper, a different distributed control method is proposed. Similar to the methods proposed in (Wang, Mahulea, and Silva 2013; Hu et al. 2014, 2015; Yang, Hu, and Liu 2015), a high-level central coordinator is not needed unlike the decentralised and hierarchical SCT architectures. In contrast to these approaches, our method is supported by automata, like basic SCT, instead of PN, and its application to a complex system is addressed.

In the proposed approach, a plant is modularly modelled according to its mechanical characteristics and the resulting local models are called plant elements (PEs). Local and global controls are treated separately. First, local safety and liveness constraints, expressed as logical Boolean equations, are applied to the corresponding local PEs according to the synthesis algorithm previously proposed (Tajer, Philippot, and Carré-Ménétrier 2013), which provides a LC for each PE. This first step was inspired by the modular SCT architecture. Second, the application of global constraints, defined in the form of logical Boolean implications, to the LCs provides distributed controllers (DCs) that allow cooperative interaction and synchronisation among the modular PEs.

One important issue regarding the supervisory control method is the implementation of the designed control. In many AMS cases, control implementation is based on programmable logic controller (PLC) which has occurred to support sequence control due to its low cost, reliability, and ease of programming (Lu and Liao 2009). Our method proposes to interpret the obtained distributed control into Grafcet[1] for purpose of PLC-based implementation. Grafcet standard (IEC Standard 60848 2013) is a language to specify the functional behaviour of control systems. If basic Grafcet is used, it is easily translated into the several PLC programming languages defined in the (IEC Standard 61131-3 2013) (David and Alla 2010). Our method provides a Grafcet specification in an automatic way, and leaves to the user the choice to transform it into a suitable IEC 61131-3 programming language (see, e.g. Schumacher and Fay 2014).

The applicability and effectiveness of the approach are validated using an experimental AMS composed of three stations, each controlled by a PLC.

Our experiment results show that the primary advantage of the proposed approach is that, it decreases the state-space explosion of the supervisory control because it avoids synchronous composition. Additionally, it ensures the flexibility required in manufacturing systems and allows two modes of operation (local and global). The integration of global constraints to the LCs allows a synchronised, cooperative functioning of the modular components while keeping their local control laws. One of this approach's objectives in the case of a redesign, is to only update and add global constraints automatically to the LCs, thus allowing PEs to be versatile in different environments.

The paper is organised as follows. Section 2 reviews some mathematical descriptions of SCT and presents some important concepts used throughout this paper. Then, the proposed control architecture is presented. Section 3 details the distributed approach concepts and provides two novel algorithms for computing the DCs as well as a method of

their Grafcet interpretation. Section 4 is devoted to the application of the approach to an experimental AMS. Section 5 summarises the results of the paper. Finally, Section 6 discusses the conclusions and offers some perspectives.

2. Preliminaries

2.1 *Supervisory control theory*

The main objective of the supervisory control theory (SCT) initiated by Ramadge and Wonham (RW) (Ramadge and Wonham 1987) was to extend the continuous systems control theory concepts to the DEDS. The RW model separates among the free behaviour of a system, modelling its entire physically realisable operation (open loop operation), and its desired behaviour (closed loop operation). In SCT, a system is assumed to evolve spontaneously, executes sequences of events that describe its behaviour, and generates a language constructed by the events alphabet. These events are divided into two disjoint sets, controllable and uncontrollable events. The objective is to synthesise a supervisor(s) which disables the occurrence of controllable events, in such a way to force the supervised system to behave according to the specifications. Formally, a DEDS is represented by the quintuple $G = (Q, \Sigma, \delta, Q_m, q_0)$, where Q is a finite set of *states*, with $q_0 \in Q$, the *initial state* and $Q_m \in Q$, the set of *marked states*, Σ is a finite set of events called an *alphabet*, and δ is a *transition function* δ: $Q \times \Sigma \rightarrow Q$.

Let Σ^* denotes the set of all finite events concatenations in Σ. An element of Σ^* is called a string. The length of a string is given by the number of its events. The empty string denoted ε is the string with no element. If u, v and ω, are strings in Σ^*, u is a prefix of ω if $uv = \omega$. A set that contains all the prefixes of all its elements is said to be *prefix closed*. A subset $L \subseteq \Sigma^*$ is called a language over Σ. The language noted $\bar{L}$containing all the prefixes of strings in L defines the prefix-closure of the language L. The language generated by G can be defined as: $L(G) = \{\omega \in \Sigma^* | \ \delta(q_0, \omega)$ is defined$\}$. $L(G)$ is also prefix-closed. The *marked language*, denoted $L_m(G) \subseteq L(G)$ is the language consisting of all strings which reach marked states. Formally, $L_m(G)$ is given by: $L_m(G) = \{\omega \in L(G) \mid \delta(q_0, \omega) \in Q_m\}$. A DEDS is said to be non-blocking if $L(G) = \overline{L_m(G)}$.

In some DEDS applications, several independent processes are considered simultaneously. To combine two DEDS (A and B) into one single more complex DEDS, i.e. $C = A \| B$, a procedure called synchronous composition is used. In the resulting automaton, common events occur synchronously, while the other events occur asynchronously (Wonham 2012). Natural projection is the inverse operation; it consists of projecting the monolithic plant on suitable local components. For $\Sigma' \subseteq \Sigma$, the natural projection P: $\Sigma^* \rightarrow \Sigma'^*$ is defined by:

$$\begin{aligned} P(\varepsilon) &= \varepsilon. \\ P(\sigma) &= \begin{cases} \varepsilon, & \text{if } \sigma \notin \Sigma' \\ \sigma, & \text{if } \sigma \in \Sigma' \end{cases} \\ P(\omega\sigma) &= P(\omega)P(\sigma), \omega \in \Sigma^*, \ \sigma \in \Sigma. \end{aligned}$$

As mentioned before, the set of events Σ is divided into two disjoint sets, the set of controllable events Σ_c, and the set of uncontrollable events Σ_{uc}. The supervisor can disable only controllable events and has no effect on uncontrollable ones. The controllability condition guarantees the existence of a supervisor. It is given by $\bar{K} \bullet \Sigma_{uc} \cap L(G) \subseteq \bar{K}$; where $L(G)$ is the physically possible behaviour and K is the desired behaviour. This condition denotes that K is controllable, if for any sequence of events ω that starts from a sequence which is already a prefix of K ($\omega \in K$), the occurrence of an uncontrollable event doesn't lead the sequence out of the desired behaviour K.

2.2 *Synthesis of local supervisory control*

In previous works (Tajer, Philippot, and V. Carré-Ménétrier 2011; Qamsane, Tajer, and Philippot 2014), we have initiated a distributed supervisory control approach that considered modular automata models for the plant and logical Boolean equations for the constraints. In this paper, we detail the steps of the approach and we refine it with formal synthesis algorithms and an implementation method. Nevertheless, we recall in this section some basics of the approach to understand the essence of this work.

2.2.1 *Plant modelling*

The construction of a plant model is primarily a complex operation when all the technology's specificities should be expressed. It is affected by state-space explosion when using a centralised approach. In our approach, we assume that the plant is divided into several modular PEs presenting all possible situations, without taking into account any

constraints originating from the control part. PEs are defined as event-driven models and use Balemi's interpretation (Balemi et al. 1993), where the set of controllable events $\Sigma_c \subseteq \Sigma$ represents the set of control outputs (actuators) and the set of uncontrollable events $\Sigma_{uc} \subseteq \Sigma$ represents the set of control inputs (sensors). We assume that the rising '↑' and falling '↓' edges associated with events are the changes of their values from 0 to 1 and from 1 to 0, respectively. According to this interpretation, the set of controllable events corresponds either to the activation orders '↑Z' or the deactivation orders '↓Z' of the control outputs; the set of uncontrollable events is associated with the rising edges '↑E' or with the falling edges '↓E' of the control inputs. Then, the sets Σ_c and Σ_{uc} are written $\Sigma_c = \uparrow Z \cup \downarrow Z$ and $\Sigma_{uc} = \uparrow E \cup \downarrow E$.

A practical model of a PE is obtained by synchronous composition of a detector model, where an initial state is imposed, with an actuator model (Philippot 2006). For each PE i, the resulting model is an automaton $G^{(PEi)} = (Q^{(PEi)}, \Sigma^{(PEi)}, \delta^{(PEi)}, q_0{}^{(PEi)})$, where $Q^{(PEi)}$ is the set of states, $\Sigma^{(PEi)}$ is the alphabet of events, $\delta^{(PEi)}$: $Q^{(PEi)} \times \Sigma^{(PEi)} \rightarrow Q^{(PEi)}$ is a transition function and $q_0{}^{(PEi)}$ is the initial state.

2.2.2 *Local safety and liveness constraints modelling*

A set of local constraints is defined for each individual PE. Two types of constraints are specified: safety constraints (what the system must not do), and liveness constraints (what the system must do). They are specified by an expert and modelled as logical Boolean equations instead of automata because of their ability to be applied locally without going through a composition step (Riera et al. 2015). Local safety and liveness constraints are represented by equations whose results must be equal to 0 (not to do) and equal to 1 (to do), respectively.

Logical equations constraints are functions that use Boolean operators {And, Or, Not} to express the consequence of events occurrence ($f((\uparrow\downarrow)e_i, (\uparrow\downarrow)z_j)$), over the activation/deactivation of the output events $(\uparrow\downarrow)z_k$ (Philippot, Tajer, and Carré-Ménétrier 2012). In this framework, the local safety and liveness constraints are modelled as follows:

$$f\left((\uparrow\downarrow)e_i,\ (\uparrow\downarrow)z_j\right) \textbf{ And } (\uparrow\downarrow)z_k{}^{\sim} = {}^{\sim}0\ (=1).$$

2.2.3 *Local controllers*

LCs are obtained by considering the local logical constraints to their corresponding PEs automata according to the local synthesis algorithm previously proposed (Tajer, Philippot, and Carré-Ménétrier 2013). The latter prohibits some controllable events from occurring to prevent the system from reaching the states not meeting the specifications. It is based on two steps. First, the local safety constraints are applied to the considered PEs, which provides local supervisors (LSUP) such as $G^{(LSUPi)} \subseteq G^{(PEi)}$, $G^{(LSUPi)} = (Q^{(LSUPi)}, \Sigma^{(LSUPi)}, \delta^{(LSUPi)}, q_0{}^{(LSUPi)})$. Second, the local liveness constraints are applied to the corresponding LSUPs, which allows to extract the LCs such as $G^{(LCi)} \subseteq G^{(LSUPi)} \subseteq G^{(PEi)}$, $G^{(LCi)} = (Q^{(LCi)}, \Sigma^{(LCi)}, \delta^{(LCi)}, q_0{}^{(LCi)})$.

2.3 *The proposed control architecture*

The proposed control architecture illustrated in Figure 1 is divided into two parts: (a) the supervisory control of a DEDS according to the SCT, and (b) the proposed approach to distributed control synthesis and implementation. The first part consists of the DEDS to be controlled; the control system; the sensor signals, considered as outputs from the DEDS and as inputs to the control system; and the control actions, considered as outputs from the control system and as inputs to the DEDS. The second part consists of the offline distributed control synthesis and implementation approach. It is based on four main steps: (i) synthesis of local control, (ii) synthesis of global control, (iii) verification of deadlock-freeness and liveness properties and (iv) interpretation of the synthesised control into Grafcet for implementation purposes. The first step aims to synthesise LCs from the plant and the local specifications models. In this step, the free behaviour of the system is modularly modelled according to the mechanical characteristics (sensors/actuators) of each PE, then, a set of local safety and liveness constraints is defined for each of these. The application of the local safety and liveness constraints to their corresponding local PEs is carried out using the local synthesis algorithm previously proposed (Tajer, Philippot, and Carré-Ménétrier 2013). The second step consists of applying the global constraints to the LCs according to the algorithms proposed in the sequel in order to obtain DCs. In the third step, a model-checking technique is used to verify that the global control satisfies the required functional properties like deadlock-freeness and liveness properties.

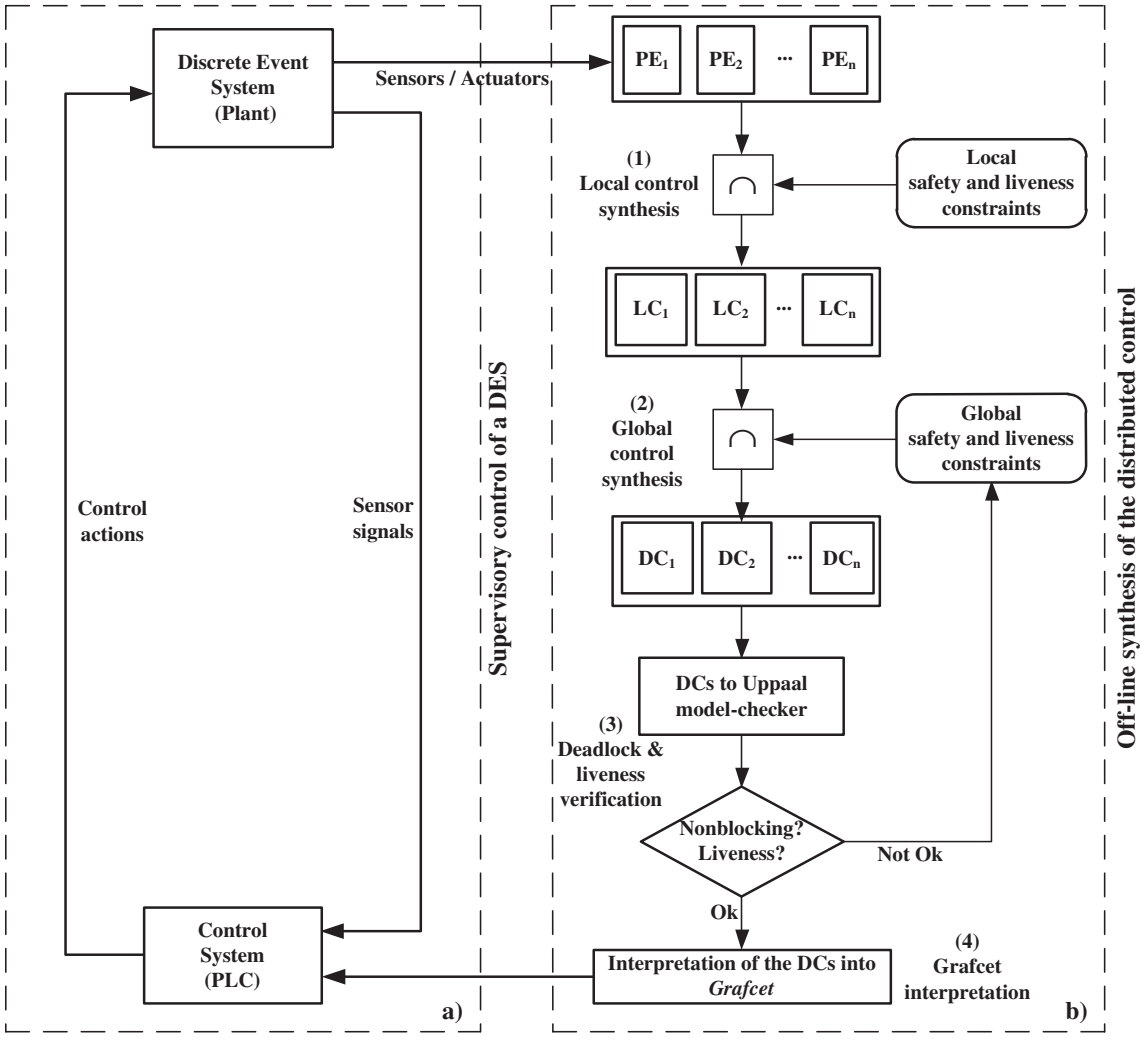

Figure 1. The proposed distributed supervisory control architecture.

The fourth step provides a method to interpret the synthesised DCs into Grafcet (IEC Standard 60848 2013) for PLC-based implementation purposes.

3. Synthesis of global distributed control

In a distributed DEDS, a subsystem is assumed to observe only the events of its locus though it may have to perform actions that depend on the other subsystems. The consideration of the global constraints to their corresponding LCs allows the subsystems to cooperate and synchronise with each other.

In this section, the proposed distributed approach is described. First, we explain how the global constraints are expressed within this framework. Second, the DC automata formalism is defined. Third, two novel algorithms of DCs synthesis are given. Fourth, we provide a method to verify that the discrete event properties: deadlock and liveness are ensured during the distributed control. Finally, a method to interpret the DCs into Grafcet is presented.

3.1 *Global safety and liveness constraints modelling*

In the proposed approach, we choose to define global constraints as logical Boolean implications. The objective is to overcome the complexity related to the modelling with automata which uses often a composition step, and also to adapt these global constraints to the proposed algorithms. In this framework, a global constraint is modelled as follows:

If (*Condition*) **Then** (*Action*).

This implication represents the consequence of a set of synchronisation events (a condition) shared among PEs over their actions. The set $C^{(spec)}$ is defined as the set of all the specifications' conditions. A condition $c \in C^{(spec)}$ could be:

(1) A *Simple condition* which consists of either a Boolean variable or a Boolean function using the symbols ↑↓, and the logic symbols ¬, ∧ and ∨, for example: ($\neg a$, $a \wedge b$, $a \vee b$, $\uparrow a$ …).
(2) A *Composed condition* that describes a sequence of Boolean variables or Boolean functions that precede each other. The symbol '→' describes the precedence rule. A composed condition is expressed by the following form: ($C_1 \rightarrow C_2 \rightarrow \ldots \rightarrow C_n$).
(3) A *Combined condition* that associates simple and composed conditions, for example: ($\uparrow a \wedge (b \rightarrow \downarrow c)$, $a \vee (b \rightarrow c)$, $(a \rightarrow b) \wedge (\downarrow c \rightarrow d)$, $(a \rightarrow b) \vee (\uparrow c \rightarrow d)$, …).

An action can be either the authorisation of a control order (order = 1) and belongs to the set $Ord^{(spec)}$, or its inhibition (order = 0) and belongs to the set $Inh^{(spec)}$. From these considerations, the set $Act^{(spec)} = \{Ord^{(spec)}, Inh^{(spec)}\}$ is defined as the set of all specifications' actions.

Definition 1: the set of global constraints is formally defined by the pair $Spec = (C^{(spec)}, Act^{(spec)})$, where $C^{(spec)}$ is the set of conditions; and $Act^{(spec)} = \{Ord^{(spec)}, Inh^{(spec)}\}$ is the set of PEs' activation/deactivation actions.

3.2 *DC automata*

In a DC automaton, controllable events are merged into macro-states as activations/deactivations of actions. The controllable events that are associated with the rising edges correspond to the authorised orders and belong to the set $Ord^{(DC)}$. Those associated with the falling edges correspond to the inhibited orders and belong to the set $Inh^{(DC)}$. The set of authorised and inhibited orders of a DC is defined as $Act^{(DC)} = \{Ord^{(DC)}, Inh^{(DC)}\}$. The set $C^{(DC)} = \{C_{Ord}{}^{(DC)}, C_{Inh}{}^{(DC)}\}$ is defined as the set of conditions that monitor the orders of $Act^{(DC)}$. $C_{Ord}{}^{(DC)}$ monitors the authorised orders, while $C_{Inh}{}^{(DC)}$ monitors the inhibited orders.

Definition 2: A DC automaton is syntactically defined by $G^{(DC)} = (Q^{(DC)}, \Sigma^{(DC)}, \delta^{(DC)}, Act^{(DC)}, C^{(DC)}, q_0{}^{(DC)})$, where $\Sigma^{(DC)}$ is a non-empty set of events such as $\Sigma^{(DC)} = \Sigma_c{}^{(DC)} \cup \Sigma_{uc}{}^{(DC)}$; $Q^{(DC)}$ is the set of states, to every state $q \in Q^{(DC)}$ is associated a set of actions $Act_q{}^{(DC)}$ (which can be empty), and a set $C_q{}^{(DC)}$ (which can be empty) of logical conditions that monitor the authorised and inhibited orders; $q_0{}^{(DC)}$ is the initial state; $Act^{(DC)} = \{Ord^{(DC)}, Inh^{(DC)}\}$ is the set of actions associated with the states of $Q^{(DC)}$; $C^{(DC)} = \{C_{Ord}{}^{(DC)}, C_{Inh}{}^{(DC)}\}$is the set of logical conditions that monitor these actions; and $\delta^{(DC)}: Q^{(DC)} \times \Sigma_{uc}{}^{(DC)} \rightarrow Q^{(DC)}$ is the transition function. A transition of $G^{(DC)}$ is defined with the triple $(q, \sigma, q') \in \delta^{(DC)}$, where q is the origin state, σ is an uncontrollable event and q' is the destination state.

Hypothesis 1: The DC automata are deterministic, that is, we cannot have two transitions with the same origin state, the same event and different destination states.

Example 1: The example of Figure 2 illustrates a DC automaton. Each node represents a state with its associated actions (authorised and inhibited orders) and conditions, if any. If a condition monitors only the authorised or inhibited orders, then, the state is divided into two parts by a line. The initial state is indicated by an incoming arrow. Each arrow

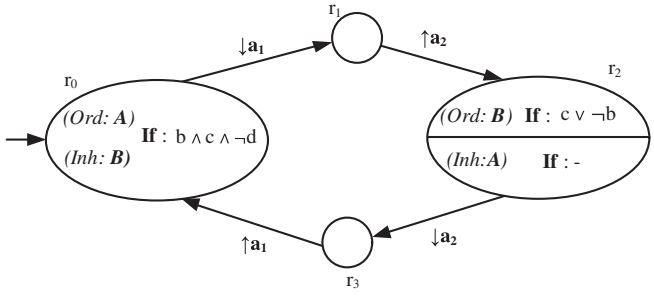

Figure 2. Example of a DC automaton.

that links two states r and r', and labelled σ ($\sigma \in \Sigma_{uc}^{(DC)}$) represents a transition (r, σ, r'). Empty actions and conditions are represented by '−'.

In the DC automaton of Figure 2, the states of $Q^{(DC)}$ are identified (r_0, r_1, r_2, r_3), and the initial state $q_0^{(DC)} = r_0$ is indicated by an incoming arrow. $\Sigma^{(DC)} = \Sigma_{uc}^{(DC)} \cup \Sigma_c^{(DC)}$, with $\Sigma_{uc}^{(DC)} = \{\downarrow a_1, \uparrow a_1, \downarrow a_2, \uparrow a_2\}$ is the set of uncontrollable events and $\Sigma_c^{(DC)} = \{\uparrow A, \downarrow A, \uparrow B, \downarrow B\}$ is the set of controllable events. $Act^{(DC)} = \{Ord^{(DC)}, Inh^{(DC)}\}$ is the set of actions associated with the states, with $Ord^{(DC)} = \{Ord_{r0} = A;\ Ord_{r1} = \varnothing;\ Ord_{r2} = B;\ Ord_{r3} = \varnothing\}$ is the set of authorised orders, and $Inh^{(DC)} = \{Inh_{r0} = B;\ Inh_{r1} = \varnothing;\ Inh_{r2} = A;\ Inh_{r3} = \varnothing\}$ is the set of inhibited orders. $C^{(DC} = \{C_{Ord}^{(DC)}, C_{Inh}^{(DC)}\}$ is the set of conditions associated with the states, with $C_{Ord}^{(DC)} = \{C_{Ordr0} = (b \wedge c \wedge \neg d);\ C_{Ordr1} = \varnothing;\ C_{Ordr2} = (c \vee \neg b);\ C_{Ordr3} = \varnothing\}$ is the set of conditions that monitor the authorised orders, and $C_{Inh}^{(DC)} = \{C_{Inhr0} = (b \wedge c \wedge \neg d.);\ C_{Inhr1} = \varnothing;\ C_{Inhr2} = \varnothing;\ C_{Inhr3} = \varnothing\}$ is the set of conditions that monitor the inhibited orders.

This automaton evolves by observing the events $\downarrow a_1$, $\uparrow a_1$, $\downarrow a_2$, $\uparrow a_2$ and executing the events $\uparrow A$, $\downarrow A$, $\uparrow B$, $\downarrow B$ as follows:

- Authorises the action A and inhibits the action B on the state r_0 while the condition $(b \wedge c \wedge \neg d)$ is fulfilled.
- When the state r_2 is reached by the occurrence of the event $\uparrow a_2$, it inhibits the action A automatically and authorises the action B, provided that the condition $(c \vee \neg b)$ is fulfilled.

Note that in a DC automaton, the occurrence of controllable events has priority over the uncontrollable ones. Indeed, uncontrollable events are consequences of controllable ones.

3.3 *Distributed control synthesis algorithms*

Two algorithms are developed for the distributed control synthesis:

(1) *Algorithm 1* (Figure 3): this algorithm provides an abstraction of the LCs. In this step, the states reached by controllable events ($z_i \in \Sigma_c$) are merged into macro-states that are connected by uncontrollable events ($e_j \in \Sigma_{uc}$). The method consists of removing the controllable evolutions from the LC, and joining them into macro-states as follows: if the controllable event is associated with a rising edge, then the order is authorised and belongs to the set $Ord^{(DC)}$; otherwise, the order is inhibited and belongs to the set $Inh^{(DC)}$. The aggregation algorithm uses the natural projection abstraction method to hide the controllable events. Because the synthesis methods require deterministic results, the algorithm avoids non-determinism after the abstraction. The DEDS software tool

Algorithm 1 Aggregation of a LC

input: $G^{(LC)} = (Q^{(LC)}, \Sigma^{(LC)}, \delta^{(LC)}, q_0^{(LC)})$
begin
1. Hide controllable events of $G^{(LC)}$
2. Determinize $G^{(LC)}$
3. Let $G^{(LACS)} = (Q^{(LACS)}, \Sigma^{(LACS)}, \delta^{(LACS)}, q_0^{(LACS)})$ be the automaton of the LC after these two operations
4. **for** every state $q \in Q^{(LACS)}$
5. **if** q merge more than one state $s \in Q^{(LC)}$ **then**
6. **for** every transition Tr between these merged states
7. Let $Tr = (s, \sigma, s)$
8. **if** $\sigma \in \uparrow Z$ **then**
9. $Ord_q^{(LAC)} \leftarrow Ord_q^{(LAC)} \cup \{z\}$
10. **elseif** $\sigma \in \downarrow Z$ **then**
11. $Inh_q^{(LAC)} \leftarrow Inh_q^{(LAC)} \cup \{z\}$
12. **endif**
13. **endfor**
14. **endif**
15. **endfor**
end
output: $G^{(LAC)} = (Q^{(LAC)}, \Sigma^{(LAC)}, \delta^{(LAC)}, Act^{(LAC)}, q_0^{(LAC))}$

Figure 3. Algorithm of the LCs aggregation.

Supremica (Akesson et al. 2006) is used to compute the natural projection and determinisation operations. The software allows saving the resulting automata in an extensible markup language (XML) (W3C 2012) file format. We denote these automata as local aggregated controllers with Supremica (LACS). The latter can be processed further as follows: check each state of the resulting controller; if a state merges more than one single state that belongs to the departure controller, then, check all the transitions between these states that were hidden by the natural projection operation; if the controllable event related to the transition is associated with a rising edge (↑), then the order is authorised and belongs to the set $Ord^{(DC)}$; if it is associated with a falling edge (↓), then it is inhibited and belongs to the set $Inh^{(DC)}$. We denote the resulting automaton as local aggregated controller (LAC). LAC is a 4-tuple $G^{(LAC)} = (Q^{(LAC)}, \Sigma^{(LAC)}, \delta^{(LAC)}, q_0{}^{(LAC)})$, where $Q^{(LAC)}$ is the set of states; $\Sigma^{(LAC)} = \Sigma^{(LC)}$, the set of events; $\delta^{(LAC)}$ is the new transition function; and $q_0{}^{(LAC)}$ is the new initial state.

(2) *Algorithm 2* (Figure 4): the objective of the algorithm 2 is to consider the global constraints to the LAC. Its principle is to check all the constraints for each LAC state. If an authorised (resp., inhibited) order of a LAC state is equal to that authorised (resp., inhibited) within a global constraint, then the constraint's condition ($C^{(spec)}$) should be associated with this state to condition the authorisation (resp., the inhibition) of the corresponding order. The resulting controller is a DC.

3.4 *Verification of deadlock-freeness and liveness properties*

The proposed approach exploits the modularity of the plant and the specifications in composite systems to avoid the computational complexity related to the synthesis of a monolithic supervisor. Asynchronous automata are used for modelling the subsystems (the PEs), whereas Boolean equations/implications are being used for modelling the local and global specifications. Nevertheless, how to guarantee the deadlock-freeness and the optimality of the global behaviour are two relevant issues, which arise when using different modelling formalisms (e.g. Automata and Boolean expressions). Such functional properties can be assessed using model-checking, which is a formal verification technique requiring a model of the system, and a desired property. It systematically checks whether the given model satisfies this property or not (Baier and Katoen 2008). Typical properties that can be checked are deadlock-freeness, invariants, and request-response properties (Liveness).

In this paper, we use the model-checker Uppaal (Behrmann, David, and Larsen 2006) to verify the deadlock-freeness and the optimality of the distributed control.

3.4.1 *Deadlock-freeness*

A deadlock occurs when the system reaches a non-marked state and no transition is defined/enabled out of that state. To check that the global control is deadlock-free using the model-checker Uppaal, we verify the property **A[] not deadlock**.

Algorithm 2 Integrating the global constraints to a LAC

input: $G^{(LAC)} = (Q^{(LAC)}, \Sigma^{(LAC)}, \delta^{(LAC)}, Act^{(LAC)}, q_0{}^{(LAC)})$.
$Spec = (C^{(spec)}, Act^{(spec)} = \{Ord^{(spec)}, Inh^{(spec)}\})$
begin
1. **for** every state $q \in Q^{(LAC)}$
2. $Ord_q{}^{(DC)} \leftarrow Ord_q{}^{(LAC)}$
3. $Inh_q{}^{(DC)} \leftarrow Inh_q{}^{(LAC)}$
4. **for** every constraint $s \in Spec$
5. **if** $Ord_q{}^{(DC)} = Ord_s{}^{(Spec)}$ **then**
6. $C_{Ordq} \leftarrow C_{Ordq} \cup \{C_s{}^{(Spec)}\}$
7. **endif**
8. **if** $Inh_q{}^{(DC)} = Inh_s{}^{(Spec)}$ **then**
9. $C_{Inhq} \leftarrow C_{Inhq} \cup \{C_s{}^{(Spec)}\}$
10. **endif**
11. **endfor**
12. **endfor**
end
output: $G^{(DC)} = (Q^{(DC)}, \Sigma^{(DC)}, \delta^{(DC)}, Act^{(DC)}, C^{(DC)}, q_0{}^{(DC)})$

Figure 4. Algorithm of the global constraints integration.

3.4.2 *Liveness*

Liveness properties are formatted as: something will eventually happen. In Uppaal, liveness properties are expressed with the path formula $\mathbf{A} < > \boldsymbol{\varphi}$, meaning $\boldsymbol{\varphi}$ is eventually satisfied or moreover with the *leads to* or *response* property, written $\boldsymbol{\varphi} \rightarrow \boldsymbol{\psi}$, which is read as whenever $\boldsymbol{\varphi}$ is satisfied, then eventually $\boldsymbol{\psi}$ will be satisfied (Behrmann, David, and Larsen 2006).

3.5 *Grafcet interpretation of the DCs*

In this part, we first present some basic concepts and notions about Grafcet and its importance in the control logic implementation. Then, the method to interpret the resulting DC automata into Grafcet is proposed.

3.5.1 *Grafcet*

Grafcet is an international standard used for logic controller specification in manufacturing systems (Zaytoon and Carré-Ménétrier 2001; Philippot and Tajer 2010; IEC Standard 60848 2013). It is a directed-graph constituted of steps and transitions. Steps are drawn as squares represent the states of the controller, to which actions can be associated to define the orders to send to the controlled part. The initial steps are described by double squares. A step can be active or idle. Actions associated with a step are performed when the step is active and remain inactive when it is idle. An action may be conditioned by a Boolean variable; in this case, its execution needs the Boolean condition to be fulfilled. A situation is specified by the set of active steps. Transitions represented as bars control the evolution of Grafcet from one situation to another, i.e. they connect one (or several) upstream step(s) to one (or several) downstream step(s). Receptivity is a logical expression associated with each transition. It allows taking into account, the Boolean variables representing the controller inputs (sensors), the activation state of individual steps, and the events corresponding to the rising and falling edges of Boolean inputs.

3.5.2 *From DC to Grafcet model*

A straightforward method for the interpretation of DCs into partial grafcets is proposed to facilitate their PLC-implementation. The method is based on the following rules:

- An ordinary DC state is represented by a step of Grafcet.
- An uncontrollable event associated with a DC transition is represented by a transition of Grafcet. The receptivity associated with the Grafcet transition is defined as the occurrence of the uncontrollable event.
- DC macro-states have different configurations depending on the authorised and inhibited orders, and their monitoring conditions. Table 1 shows the possibilities of state configurations existence. For example, in the configuration 1, no order is authorised or inhibited and no monitoring conditions are set, which means that this configuration

Table 1. Different configuration possibilities of the DC's macro-states.

id	*Ord*	*Inh*	C_{Ord}	C_{Inh}	State existence
1	0	0	0	0	1
2	0	0	0	1	∅
3	0	0	1	0	∅
4	0	0	1	1	∅
5	0	1	0	0	1
6	0	1	0	1	1
7	0	1	1	0	∅
8	0	1	1	1	∅
9	1	0	0	0	1
10	1	0	0	1	∅
11	1	0	1	0	1
12	1	0	1	1	∅
13	1	1	0	0	1
14	1	1	0	1	∅
15	1	1	1	0	1
16	1	1	1	1	1

corresponds to an ordinary state. Note that it is not possible to have conditions in a macro-state without their corresponding authorised and inhibited orders. This makes the states existence in configurations 2, 3, 4, 7, 8, 10 and 12 equal to ∅.

- When an order is authorised and another is inhibited in a same DC macro-state, we assume that the priority goes to the inhibition i.e. one must disable the command already activated before enabling the authorised one to ensure safety. In state configuration 14 one can see that the inhibited order is conditioned, while the authorised one is not, this configuration is not allowed (equals ∅) because it gives priority to the authorisation over inhibition, which contradict the assumption above.

Altogether 11 interpretation rules have been defined to ensure a correct transformation of the DCs into Grafcet, see Figures 5 and 6.

Note that the type of conditions can affect the Grafcet structure. Simple conditions can be added directly to the corresponding transitions. Nevertheless, both composed and combined conditions expand the Grafcet structure. For example, the condition ($C_1 \rightarrow C_2 \rightarrow C_3$), composed of three succeeding conditions, will replace a simple transition by the structure of Figure 7.

4. Application to an experimental manufacturing system

4.1 *The experimental manufacturing system*

The experimental manufacturing system shown in Figure 8 comprises three stations distributing, testing and sorting three types of workpieces (black, red and silver). Each station is controlled by a PLC Siemens S7-300.

The distributing station is divided into three PEs: A feeder (single-acting cylinder), a swivel drive (double-acting cylinder) and a suction cup. The feeder feeds the workpieces from a buffer to the pickup area; the swivel drive transports the workpieces towards the testing station; and the suction cup catches the workpieces while the swivel drive is moving. The testing station is divided into three PEs: an elevator (double-acting rodless cylinder), an ejector (single-acting cylinder) and an air cushioned slide. The elevator lifts the workpieces towards a measuring module that checks their height; the ejector pushes out the compliant workpieces to the sorting station through the upper air cushioned slide, and ejects the non-compliant workpieces to the lower slide. The sorting station is composed of four PEs: a conveyor, a barrier (single-acting cylinder) and two flippers (single-acting cylinder). The conveyor belt, actuated by a permanent magnetic direct current motor, transports the workpieces towards three sorting slides; the pneumatic barrier blocks the workpieces at the entrance of the station to identify their colour and material; and the pneumatic flippers are used to select the slides, where the workpieces should be sorted. Each station uses a set of sensors that indicate the arrival of workpieces, their colour and material, working areas freeness and the stations' statutes (occupied/vacant). The communication between the PLCs is carried out using SEND/RECEIVE protocol.

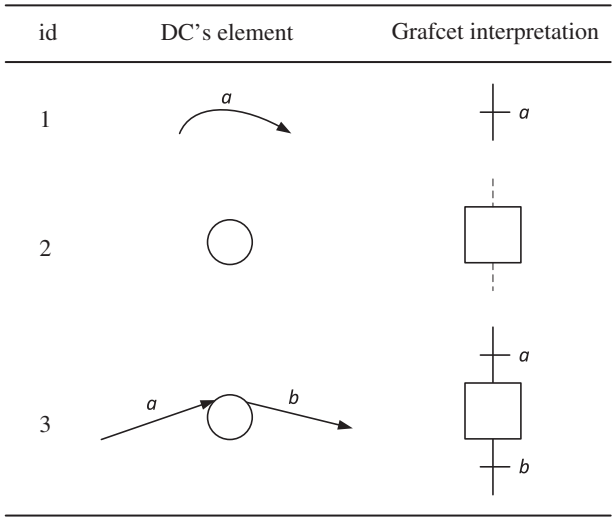

Figure 5. Rules of Grafcet interpretation for the DC simple evolutions.

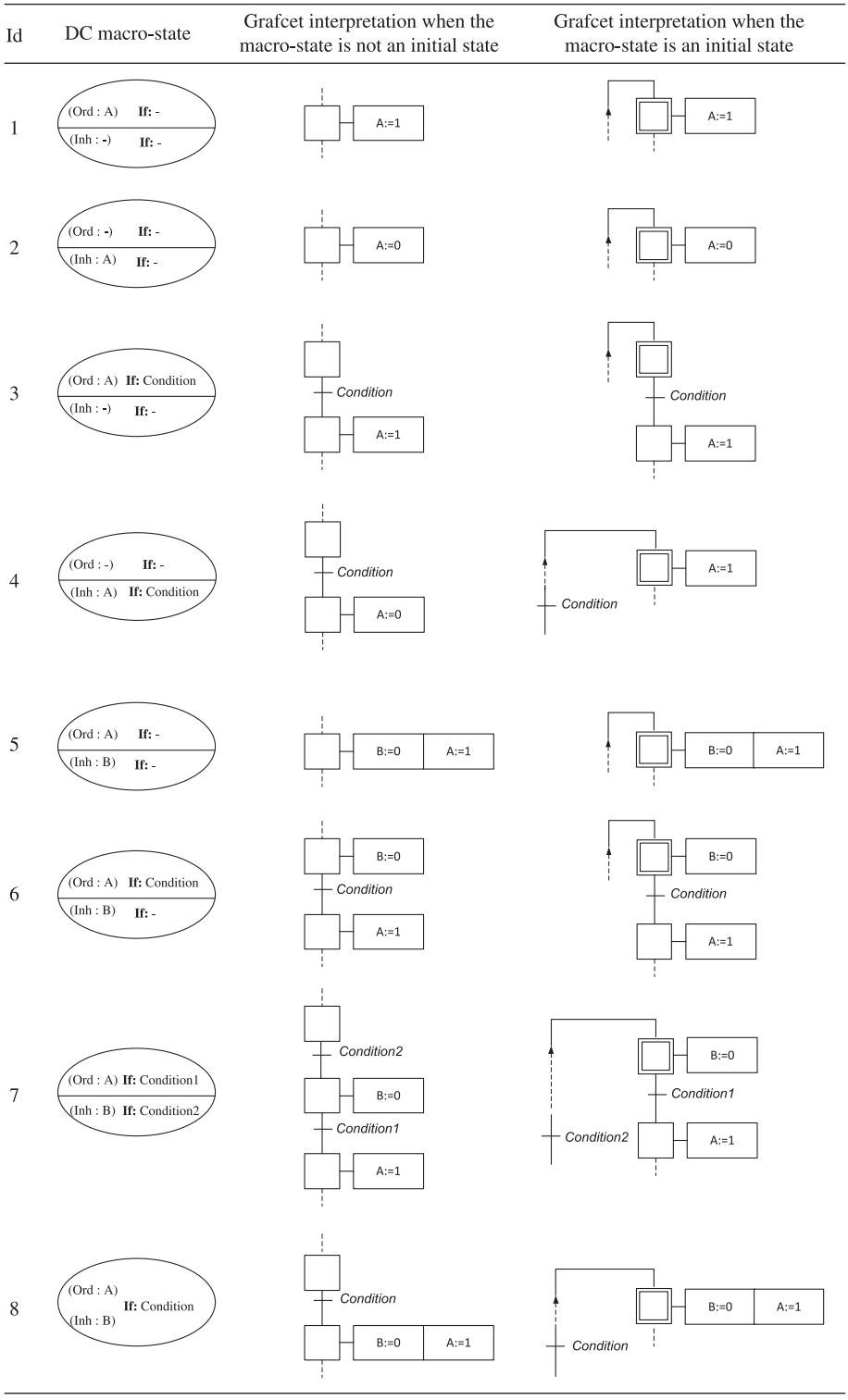

Figure 6. Rules of Grafcet interpretation for the DC macro-states.

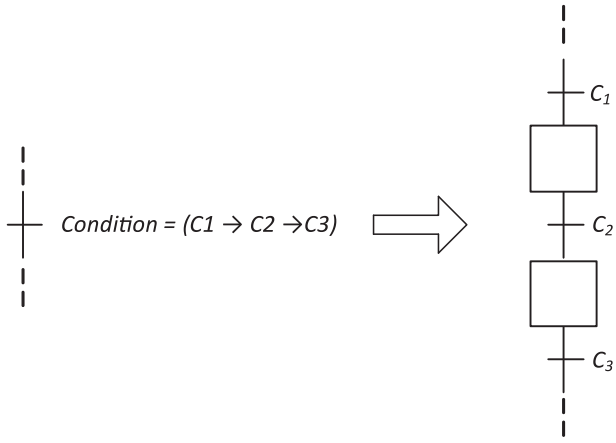

Figure 7. Example of a composed condition's Grafcet interpretation.

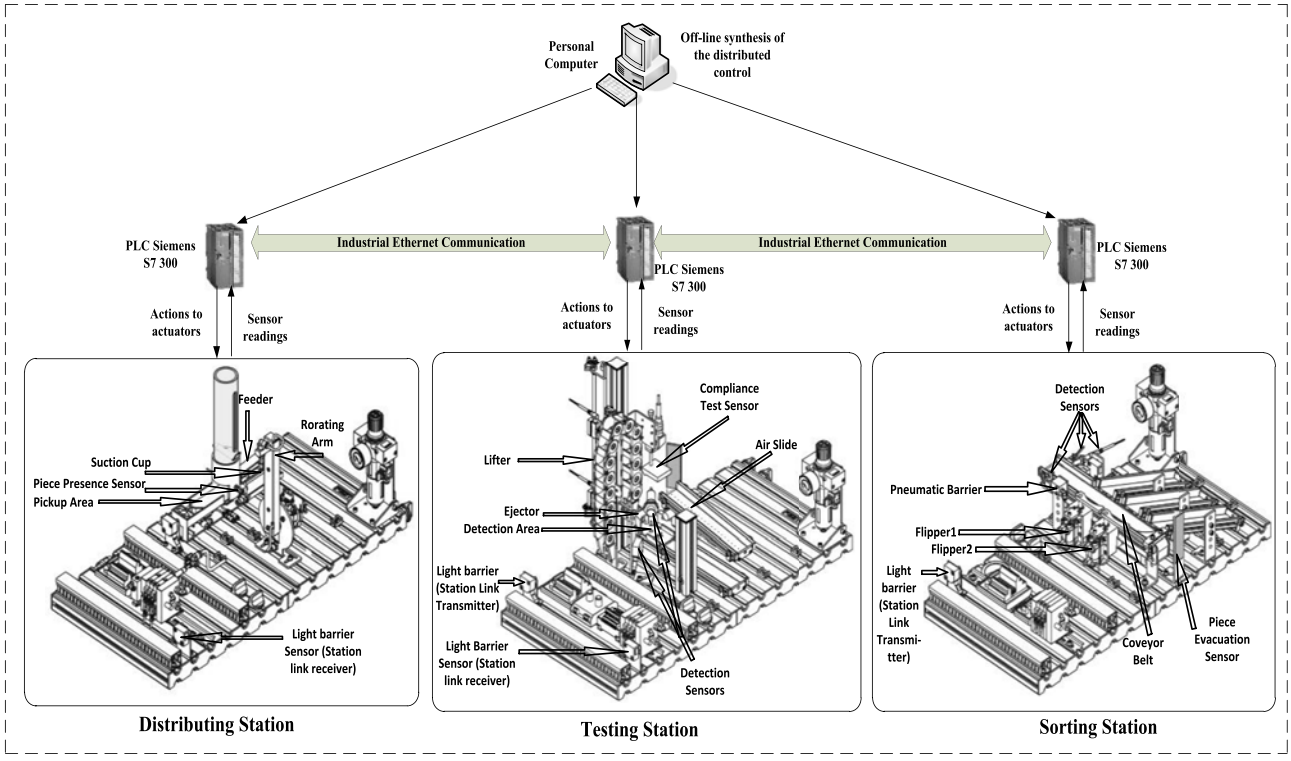

Figure 8. The experimental manufacturing system.

We detail and illustrate the steps of the approach by means of the distributing station, and in the final step we give the three stations global control by applying the same principles to the other two stations.

Tables 2 and 3 represent descriptions of the distributing station components and their event labels.

4.2 *Plant modelling*

As mentioned above, the distributing station is constituted of three PEs (feeder, swivel drive and suction cup). PEs models are given in Figure 9. We do not explain their construction. The reader can find detailed explanation in (Philippot 2006; Philippot, Sayed Mouchaweh, and Carré-Ménétrier 2009).

Table 2. PLC1 inputs.

	Sensor	Description	Event label
Distributing station	Through-beam sensor	Detects if a workpiece is available	wpa1
	Proximity sensor	Detects if the feeder is in back position	fbp
	Proximity sensor	Detects if the feeder is in front position	ffp
	Limit switch	Detects if the swivel drive is in magazine position	dmp
	Limit switch	Detects if the swivel drive is in testing station position	dsp
	Pressure switch	Detects if a workpiece is caught	wpc
	Station link receiver	Detects if the testing station is vacant	testing_vacant

Table 3. PLC1 outputs.

	Actuator	Description	Event label
Distributing station	Feeder	The feeder extends	Feed_ext
	Swivel drive	The swivel drive moves to the magazine position	Go_mag
		The swivel drive moves to the testing station position	Go _stn
	Suction cup	The suction switches on	CatchUp
		The suction switches off	Release

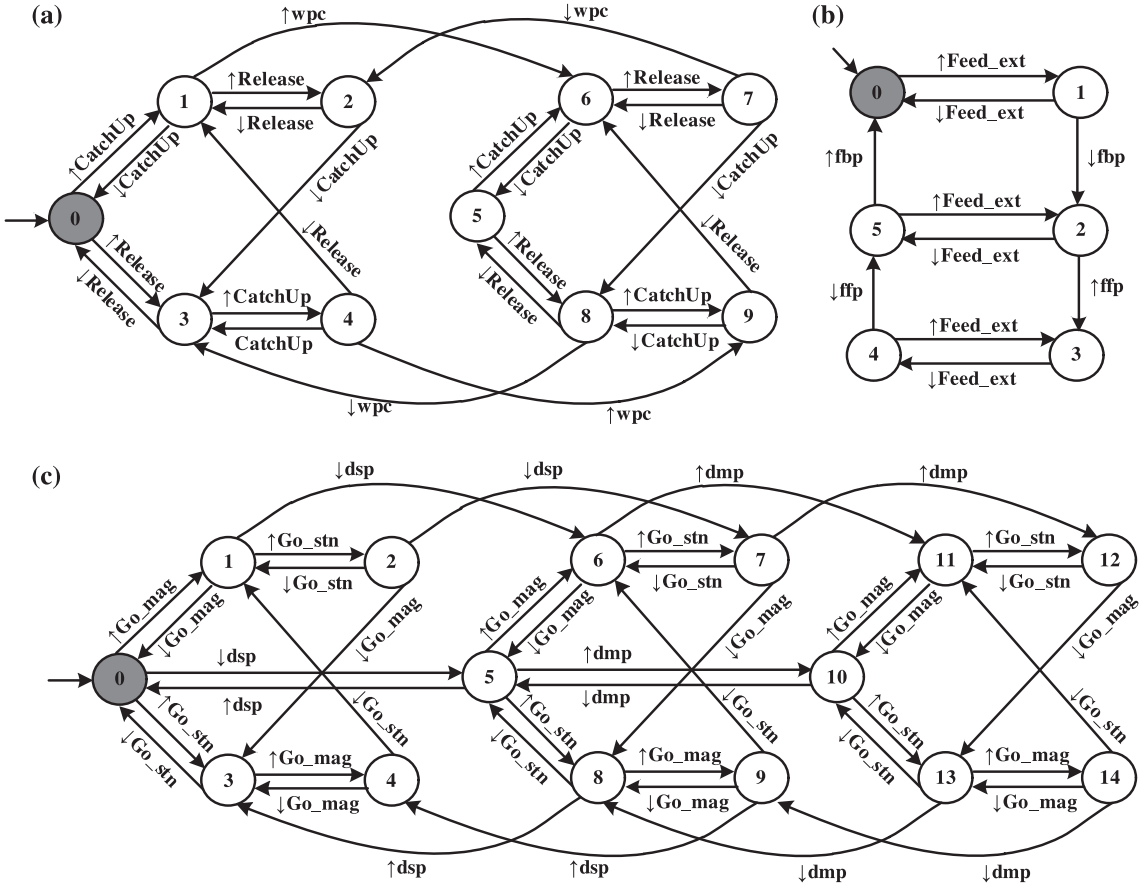

Figure 9. The PE models. (a) suction cup, (b) feeder, (c) swivel drive.

Table 4. Local safety and liveness constraints.

PE	Type	Constraints
Swivel drive	Safety	↑Go_stn **And** ↑Go_mag = 0
		Not ↑Go_stn **And Not** ↑Go_mag **And Not** ↑dmp **And Not** ↑dsp = 0
	Liveness	↑Go_stn **And** ↓dmp = 1
		↑Go_mag **And** ↓dsp = 1
		↓Go_stn **And** ↑Go_mag = 1
		↓Go_mag **And** ↑Go_stn = 1
Feeder	Safety	↓fbp **And** ↓Feed_ext = 0
		↓ffp **And** ↑Feed_ext = 0
	Liveness	↑Feed_ext **And** ↓fb*p* = 1
		↓Feed_ext **And** ↓ff*p* = 1
Suction cup	Safety	↑CatchUp **And** ↑Release = 0
	Liveness	↑CatchUp **And** ↑wpc = 1
		↓CatchUp **And** ↑Release = 1
		↓CatchUp **And** ↑Release = 1
		↑Release **And** ↓wpc = 1

4.3 *Local safety and liveness Constraints*

Table 4 presents the local safety and liveness constraints (obtained by an expert) to apply to the distributing station PEs. For example, the safety constraint '**Not** ↑Go_stn **And Not** ↑Go_mag **And Not** ↑dmp **And Not** ↑dsp **= 0**' reflects the fact that the swivel drive should not be placed in an intermediate position, where none of the orders 'Go_stn' and 'Go_mag' are activated and none of the sensors 'dmp' and 'dsp' are reached.

4.4 *Local controllers*

The application of the safety and liveness constraints presented in Table 4 to the corresponding PEs according to the local synthesis algorithm previously proposed (Tajer, Philippot, and Carré-Ménétrier 2013), allows obtaining the LCs of Figure 10.

4.5 *Global constraints*

Global constraints allow a cooperative functioning among the subsystems forming the global system. Ten specifications are stipulated for the distributing station as follows: (1) The feeder doesn't extend if the swivel drive is in the magazine position; (2) The feeder doesn't extend without a workpiece; (3) The feeder doesn't extend while the swivel drive is moving to the magazine position; (4) The swivel drive only goes to the magazine position when necessary (i.e. the workpiece is deposited to the testing station, and the feeder is in the front position); (5) The swivel drive goes to the testing station position only if the latter is vacant; (6) The swivel drive goes to the testing station when a workpiece is securely caught; (7) Mutual exclusion between the suction cup activation, and the swivel drive goes to the magazine position; (8) The suction cup doesn't activate without a workpiece in the pick-up area; (9) The feeder retracts when a workpiece is transported by the swivel drive; (10) The suction cup releases the workpiece when the swivel drive reaches the testing station position.

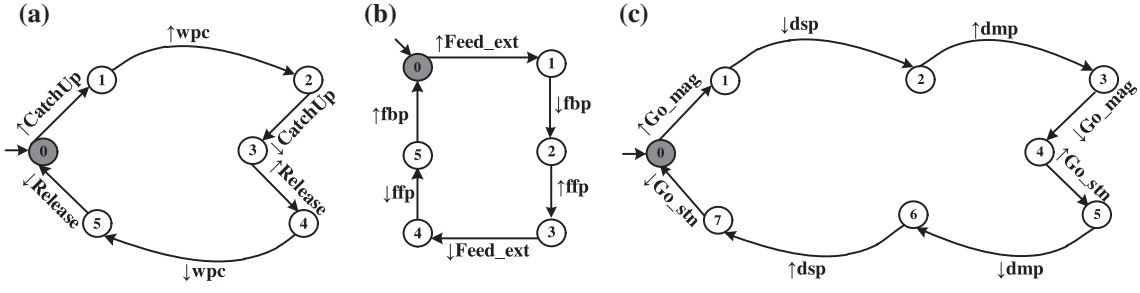

Figure 10. The LC models. (a) suction cup, (b) feeder, (c) swivel drive.

For example, the first three constraints are related to the authorisation of the order: feeder extend. These constraints allow the feeder to extend when the drive is not in the magazine position (¬*dmp*), a workpiece is detected in the buffer (*wpa1*), and the swivel drive is not moving to the magazine position (¬*Go_mag*), respectively.

As explained in Section 3.1, this can be written in Boolean algebra as follows: **if** (¬*dmp*∧¬*Go_mag*∧*wpa1*) **then** *Ord(Feed_ext)*.

Table 5 represents the set of global constraints developed from the ten informal specifications above for the distributing station.

4.6 *Distributed controllers*

To synthesise the experimental system DCs, we use the proposed algorithms. First, the LC automata are fed to the software Supremica, where the natural projection operation is carried out. Each LC is modelled as a state machine using the command 'New automaton'. For each LC, we hide the set of controllable events (natural projection operation). For this task, the commands 'Hide events' and 'Minimise' are used. Second, we export each automaton as an XML file and complete it by adding to each state, the sets of authorised and inhibited orders, if any, according to algorithm 1.

For illustration, the XML files of the feeder are shown in Figure 11. By hiding the controllable events: ↑*Feed_ext* (*up_Feed_ext*) and ↓*Feed_ext* (*down_ Feed_ext*) of the feeder's LC (Figure 11(a)), we obtain its corresponding LACS

Table 5. Global constraints.

		Then	
	If	*Ord*	*Inh*
Distributing station	¬dmp∧wpa1∧¬Go_mag	Feed_ext	
	¬wpc∧ffp∧¬CatchUp	Go_mag	
	wpc∧testing_vacant	Go_stn	
	¬Go_mag∧(ffp → dmp)	CatchUp	
	dsp	Release	CatchUp
	dmp∧wpc		Feed_ext

Figure 11. Aggregation steps. (a) XML representation of the feeder's LC, (b) XML representation of the feeder's LACS.

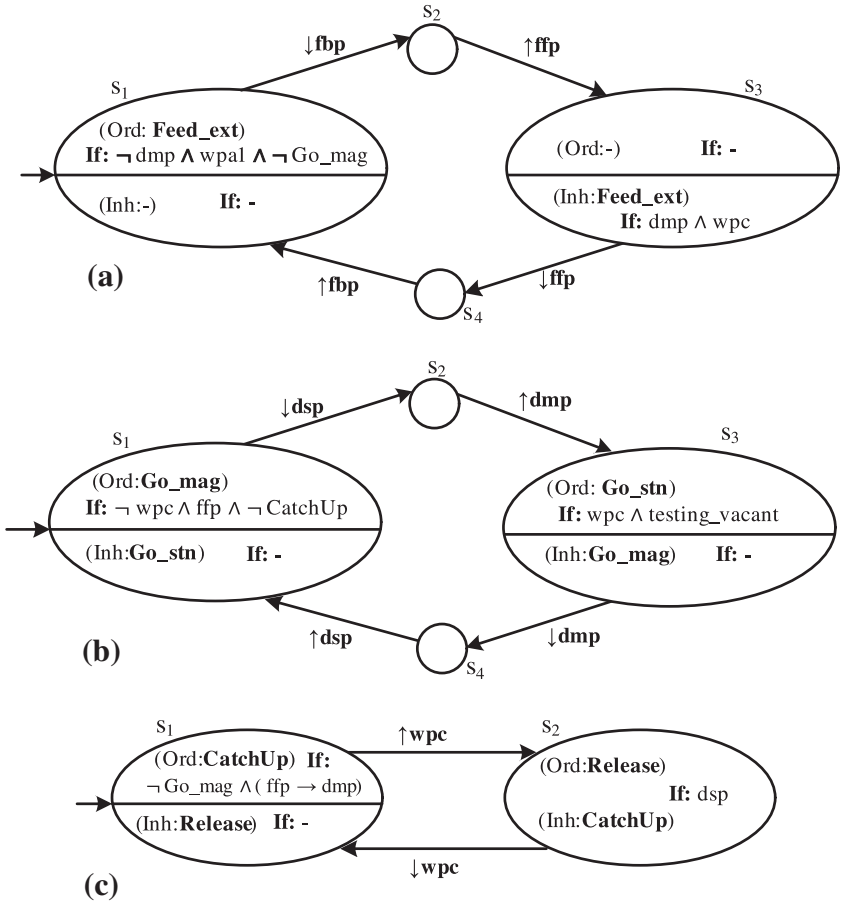

Figure 12. The DC models. (a) feeder, (b) swivel drive, (c) suction cup.

(Figure 11(b)), where one can see that the initial state (s_0) merges the states s_0 and s_1 of the departure automaton; Algorithm 1 checks the transitions between the states s_0 and s_1 of the latter. The event associated with the corresponding transition is of id = 3, which relates to the event *up_ Feed_ext* (↑*Feed_ext*). Because this event is associated with a rising edge (up (↑)), it belongs to the set *Ord* in the macro-state s_0 of the feeder's LAC. Similarly, the event *down_ Feed_ext* (↓*Feed_ext*) belongs to the set *Inh* in the macro-state s_2.

As per Table 5, the global constraints which relate to the feeder are: (i) **if** (¬*dmp*∧¬*Go_mag*∧*wpa1*) **then** *Ord(Feed_ext)* and (ii) **if** (*dmp*∧*wpc*) **then** *Inh(Feed_ext)*. Algorithm 2 adds the condition (¬*dmp*∧¬*Go_mag*∧*wpa1*) to the macro-state s_0 of the feeder's LAC, where the order *Feed_ext* is authorised (∈ *Ord*), and adds the condition (*dmp*∧*wpc*) to the macro-state s_2, where the order *Feed_ext* is inhibited (∈ *Inh*). The resulting DC is given in Figure 12(a).

In a similar way, we obtain the DCs of the swivel drive (Figure 12(b)), and the suction cup (Figure 12(c)).

4.7 *Verification of deadlock-freeness and liveness properties*

In this step, the obtained DCs are fed to Uppaal software in order to verify the deadlock-freeness and liveness properties. Figure 13 shows the automata representing the DCs of the distributing station within Uppaal software.

Uppaal indicates that the global control is deadlock-free and the DCs embody the system's maximally permissive supervised behaviour within the specifications.

4.8 *Implementation of the DCs*

In this final step, the DCs are interpreted as partial grafcets for purpose of PLC-based implementation using the interpretation rules of Figures 5 and 6.

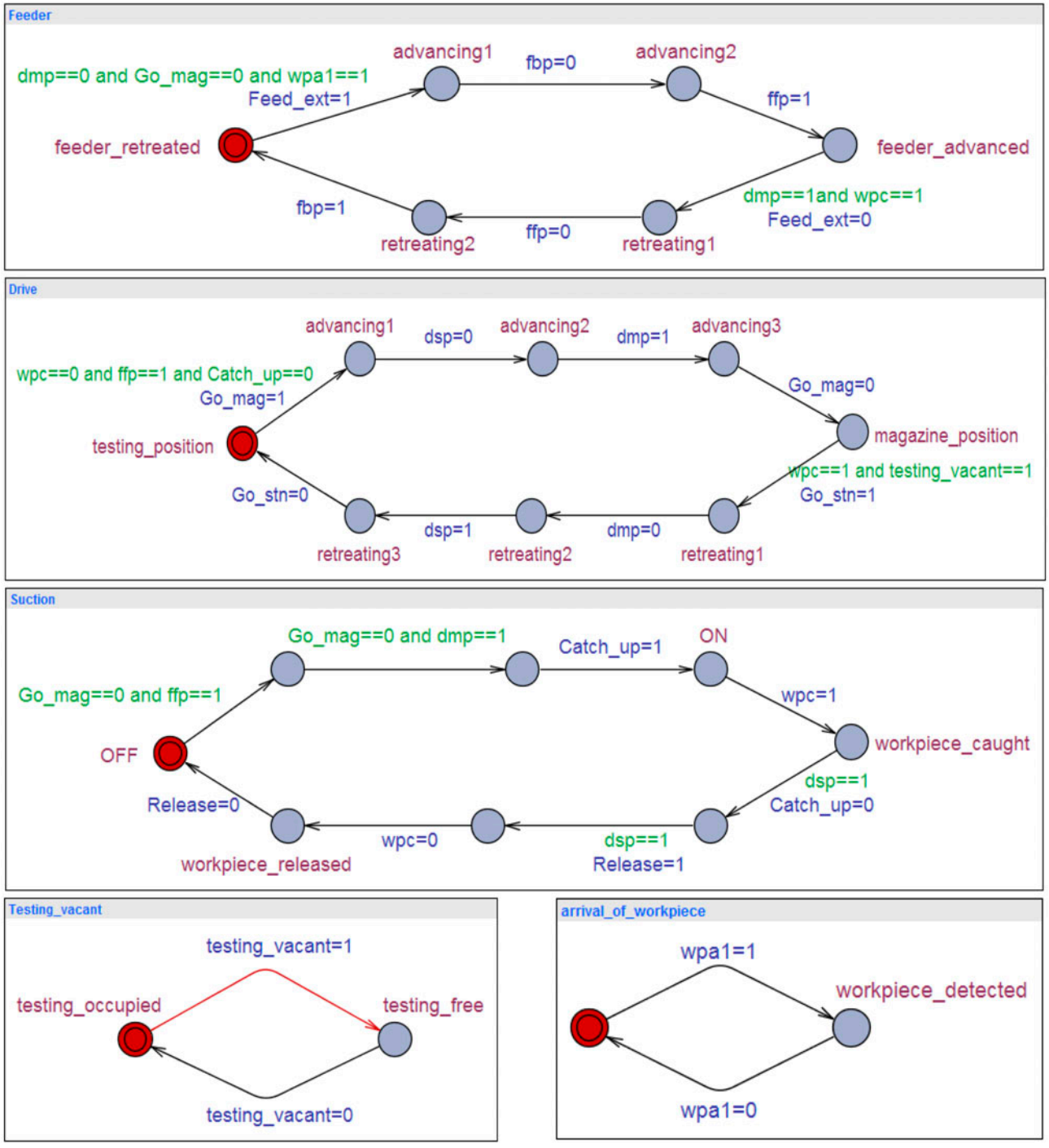

Figure 13. The DCs of the distributing station within Uppaal software.

For example, the grafcet of Figure 14(a) corresponds to the feeder DC. In this DC (Figure 12(a)), the states s_1 and s_3 correspond to the interpretations 3 and 4 of Figure 6, respectively. The two remaining evolutions correspond to the interpretation 3 of Figure 5.

Figure 14 shows the partial grafcets that correspond to the overall experimental system, which are obtained in a similar way. These grafcets are tested on the experimental system, and the observed behaviour doesn't violate the desired behaviour.

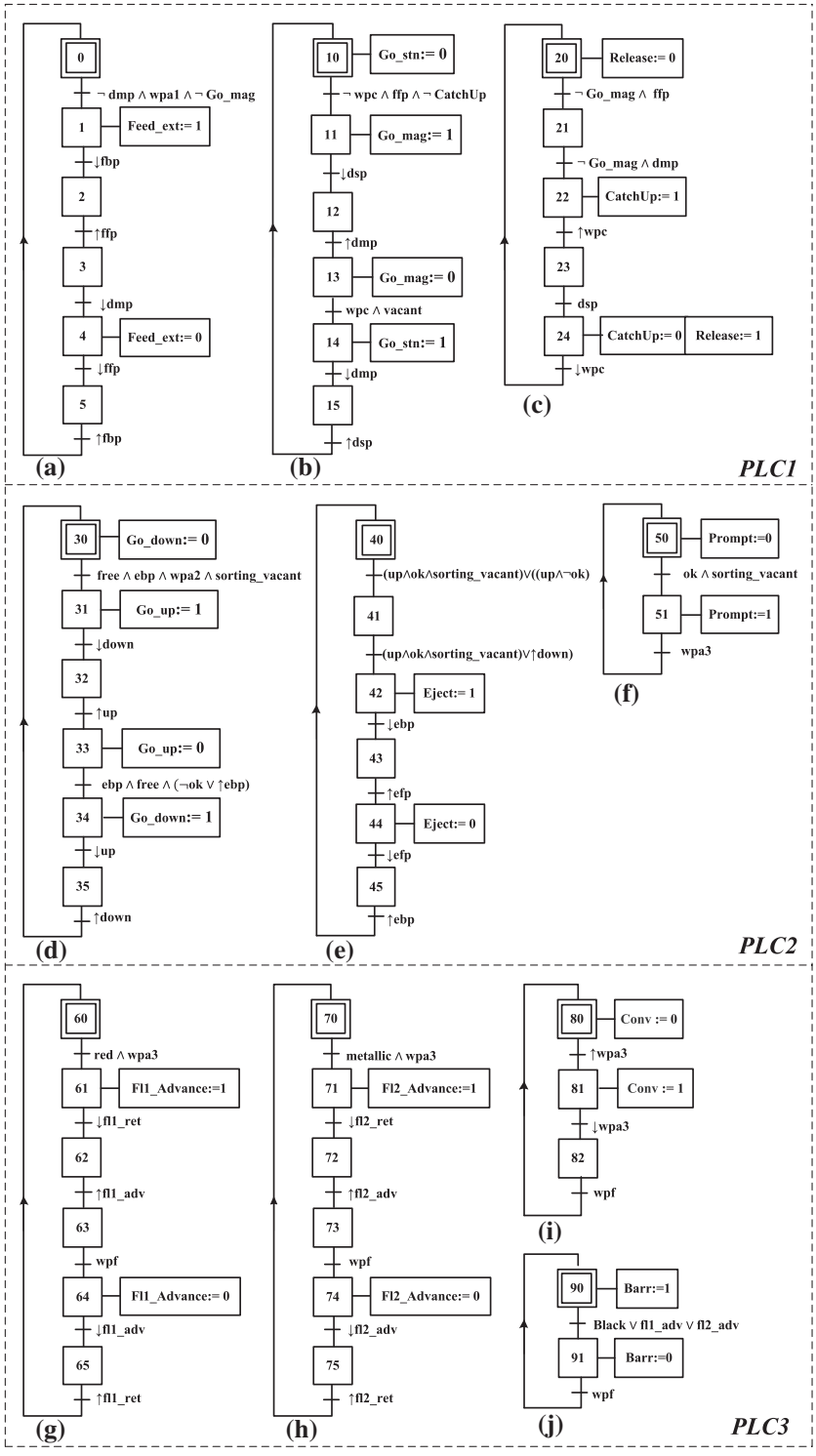

Figure 14. The Grafcet models. (a) feeder, (b) swivel drive, (c) suction cup, (d) elevator, (e) ejector (f) cushioned air slide, (g) first flipper, (h) second flipper, (i) conveyor, (j) barrier.

Table 6. Comparison between the centralised and the proposed distributed approach in terms of states and transitions numbers.

		Centralised Monolithic system	Distributed		
			Feeder	Swivel drive	Suction cup
States	Plant	900	6	15	10
	Controller	324	4	4	2
	Grafcet	–	6	6	5
Transitions	Plant	6180	10	42	24
	Controller	1080	4	4	2
	Grafcet	–	6	6	5

5. Discussion

The main cause of computational challenges in a given system G is when G is the result of a parallel composition of many subsystems. In the worst case, state space size of $G = \|_{i=1}^{n} G_i$ grows exponentially with n, the number of subsystems (Cassandras and Lafortune 2008).

Our approach is based on a modular structure and doesn't use synchronous composition to couple the subsystems. Their coupling is obtained by adding conditions (expressions of events) that monitor authorisations and inhibitions of subsystem actions to their DCs macro-states.

For standard supervisory control, the number of states is proportional in the worst case to $M^k 2^{\Sigma c+\Sigma u}$, where M is the number of states in the constraint automaton and k the number of constraints. Otherwise, integrating the constraints modelled as logical equations to the PE automaton, the number of states is proportional in the worst case to $2^{\Sigma c+\Sigma u}$.This significant decrease is because synchronous composition between the process and the constraints models is avoided. In the worst case for local and global control, the size of each controller is that of its corresponding modular PE automaton.

Table 6 provides to make a comparison between the total number of states in the plant, the supervisors and the controllers for the distributing station when the centralised and the proposed distributed approaches are used.

Table 6 shows that the controllers obtained with the distributed approach have fewer state numbers compared to the monolithic controller. This results in an easy computation and understanding of the models and also in less memory use for implementation. The proposed approach allows implementation of concise modular DCs and attainment of optimal closed-loop behaviour with their interpretation into three partial grafcets (Figure 14(a)–(c) for the distributing station) that function concurrently, instead of a centralised controller. (Note that Grafcet interpretation is not performed within the centralised approach because of the computational complexity).

The obtained DC models are enriched models, where all information (activations and deactivations) of sensors and actuators are presented. For example, the feeder model (Figure 12(a)) contains a state between deactivation of the sensor 'fbp' and activation of the sensor 'ffp' (state s_2 of the feeder DC). The action Feed_ext must be maintained until the state s_3 is reached, which justifies the choice of using stored actions instead of continuous actions in the DC Grafcet interpretation.

For control purposes, the states between sensor activations and deactivation can be hidden by a natural projection without affecting the desired command. This could be a subject of a reduction step to improve and optimise the Grafcet control models in future research.

To measure the performance of the approach, a comparison between Grafcet intuitively obtained by several students, and Grafcet obtained by the proposed approach for the distributing system has been made. Most students have realised one single linear grafcet. Compared to the three partial grafcets obtained by the proposed approach, the students' grafcet is relatively optimal in term of steps number. It contains twelve steps, while the partial grafcets obtained by the proposed approach contain a total of fourteen steps. Otherwise, the proposed approach gives more flexibility to the components constituting the overall system. Flexibility is gained because the components operate concurrently, and don't have to wait for the operating of each other as in a linear structure. Moreover, the production with the proposed approach was very optimal compared to the students' control in terms of the treated workpiece numbers during one hour of operation.

6. Conclusion and prospects

This paper outlines a distributed control synthesis approach for AMS. The approach is based on the use of modular plant models, and Boolean equations to model their corresponding local desired behaviour. The aim is to obtain

abstracted models easier to exploit, especially in case of complex systems in order to overcome the state-space explosion issue. LCs are obtained for each individual subsystem, then, global constraints are requested to cooperatively execute control actions. DCs are obtained by integrating the global constraints to the corresponding LCs. DCs are simple, adaptive (in the case of a redesign, a small amount of data will be updated), and reduce the computational problem. For PLC-based implementation purposes, a method to interpret the DCs into Grafcet is proposed. Finally, an experimental manufacturing system is used to demonstrate the applicability and effectiveness of the approach in industrial control applications.

The key advantage of this approach is the use of logical Boolean equations/implications instead of automata to model the subsystems individual specifications and interaction dependencies, which avoids computational complexity. Because this modelling is not familiar to the control users (it is made by an expert), let us also mention that the reduction of the control problem comes at the cost of a lower visibility and understanding of the specifications. This leaves an open question about the way informal specifications are modelled. The subsequent modelling problem arises: How do we know that this modelling correctly reflects the original informal specifications? Future applications work would consider this problem.

Another interesting problem to be considered is how the deadlock-freeness and liveness properties are guaranteed by the approach. Indeed, because Grafcet is similar to control interpreted PN, it would be possible to use a combined Grafcet for all the DCs, then, use graphical methods to ensure these properties. But establishing a combined Grafcet for the DCs also requires an appropriate method. This will take more effort to cope with problems in case properties are violated, i.e. one must go up to refine the global constraint leading to the problem, then refine the DCs, and re-establish a combined Grafcet model and so forth. Otherwise, the authors use a model-checker as a filter which verifies whether the defined global constraints don't lead to deadlocks or violate the system liveness before the implementation level. The advantage of the model-checker is to show the trace leading to the problem, which allows swift reaction to the global constraints conducting to the latter, then, swift update of the DCs. The Grafcet interpretation only comes once the DCs satisfy the requested properties. A strategy to refine the global constraints, in case the deadlock-freeness and liveness properties aren't guaranteed, will be addressed in future work. We intend also to refine the approach by adding a DC reduction step to optimise the grafcets as mentioned in the previous section. Moreover, we expect to develop a control synthesis tool for AMS, based on the proposed approach.

Disclosure statement

No potential conflict of interest was reported by the authors.

Note

1. Grafcet will be mentioned with a capital G when referring to the tool in general, and mentioned with a small g (grafcet) when referring to a specific logic controller model.

References

Akesson, K., M. Fabian, H. Flordal, and R. Malik. 2006. "Supremica-an Integrated Environment for Verification, Synthesis and Simulation of Discrete Event Systems." In *IEEE 8th International Workshop on Discrete Event Systems*, 384–385. Michigan.

Baier, C., and J. P. Katoen. 2008. *Principles of Model Checking*. Boston, MA: MIT Press.

Balemi, S., G. J. Hoffmann, P. Gyugyi, H. Wong-Toi, and G. F. Franklin. 1993. "Supervisory Control of a Rapid Thermal Multiprocessor." *IEEE Transactions on Automatic Control* 38 (7): 1040–1059.

Behrmann, G., A. David, and K. G. Larsen. 2006. "A Tutorial on Uppaal." In Formal Methods for the Design of Real-time Systems, 200–236. Berlin Heidelberg: Springer-Verlag.

Biallas, S., J. Brauer, and S. Kowalewski. 2012. "Arcade.PLC: A Verification Platform for Programmable Logic Controllers." In *Proceedings of the 27th IEEE/ACM International Conference on Automated Software Engineering, ASE*, 338–341. Essen.

Cassandras, C. G., and S. Lafortune. 2008. *Introduction to Discrete Event Systems*. 2nd ed. New York: Springer.

Cury, J. E. R., M. H. de Queiroz, G. Bouzon, and M. Teixeira. 2015. "Supervisory Control of Discrete Event Systems with Distinguishers." *Automatica* 56: 93–104.

David, R., and H. Alla. 2010. *Discrete, Continuous, and Hybrid Petri Nets*. Berlin: Springer.

Feng, L., and W. M. Wonham. 2008. "Supervisory Control Architecture for Discrete-event Systems." *IEEE Transactions on Automatic Control* 53: 1449–1461.

Gouyon, D., J. F. Petin, and A. Gouin. 2004. "Pragmatic approach for modular control synthesis and implementation." *International Journal of Production Research* 42 (14): 2839–2858.

Hill, R. C., J. E. R. Cury, M. H. de Queiroz, D. M. Tilbury, and S. Lafortune. 2010. "Multi-level Hierarchical Interface-based Supervisory Control." *Automatica* 46 (7): 1152–1164.

Hopcroft, J., R. Motwani, and J. Ullman. 2006. *Introduction to Automata Theory, Languages, and Computation*. 3rd ed. Boston, MA: Addison-Wesley Longman.

Hu, H., C. Chen, R. Su, Y. Liu, and M. C. Zhou. 2014. "Distributed Supervisor Synthesis for Automated Manufacturing Systems Using Petri Nets." In 2014 IEEE International Conference on Robotics and Automation (ICRA), 4423–4429. Hong Kong.

Hu, H., R. Su, M. C. Zhou, and Y. Liu. 2015. "Polynomially Complex Synthesis of Distributed Supervisors for Large-scale AMSs Using Petri Nets." *IEEE Transactions on Control Systems Technology* 24 (5): 1–13.

IEC Standard 60848. 2013. *Grafcet Specification Language for Sequential Function Charts*. International Electrotechnical Commission. Geneva.

IEC Standard 61131-3. 2013. *Programmable Controllers – Part 3: Programming Languages*. International Electrotechnical Commission. Geneva.

Komenda, J., T. Masopust, and J. H. van Schuppen. 2012. "Supervisory Control Synthesis of Discrete-event Systems using a Coordination Scheme." *Automatica* 48 (2): 247–254.

Lu, M. S., and S. F. Liao. 2009. "An Integrated IDEF0-3/CTPN/SFC Approach for Design and Analysis of Discrete Event Control Systems." *International Journal of Production Research* 47: 6433–6453.

Philippot, A. 2006. "Contribution Au Diagnostic Décentralisé Des Systèmes À Événements Discrets: Application Aux Systèmes Manufacturiers." PhD thesis, Université de Reims-Champagne Ardenne (in French).

Philippot, A., and A. Tajer. 2010. "From GRAFCET to Equivalent Graph for Synthesis Control of Discrete Events Systems." In *The IEEE 18th Mediterranean Conference on Control and Automation (MED10)*, 683–688. Marrakech.

Philippot, A., M. Sayed Mouchaweh, and V. Carré-Ménétrier. 2009. "Modelling of a Discrete Manufacturing System by Parts of Plant." In *13th IFAC Symposium on INformation COntrol Problems in Manufacturing (INCOM'09)*, 343–348. Moscow.

Philippot, A., A. Tajer, and V. Carré-Ménétrier. 2012. "From Centralized to Decentralized Approach for Optimal Controller of Discrete Manufacturing Systems." *ARPN Journal of Science and Technology* 2 (10): 936–949.

Qamsane, Y., A. Tajer, and A. Philippot. 2014. "Synthesis and Implementation of Distributed Control for a Flexible Manufacturing System." In *IEEE Second World Conference on Complex Systems (WCCS)*, 323–329. Agadir.

Ramadge, P. J., and W. M. Wonham. 1987. "Supervisory Control of a Class of Discrete Event Processes." *SIAM Journal on Control and Optimization* 25 (1): 206–230.

Riera, B., A. Philippot, R. Coupat, F. Gellot, and D. Annebicque. 2015. "A Non-intrusive Method to Make Safe Existing PLC Program." *The 9th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes (SAFEPROCESS'15)*. Paris.

Sayed-Mouchaweh, M., A. Philippot, and V. Carre-Menetrier. 2008. "Decentralized Diagnosis Based on Boolean Discrete Event Models: Application on Manufacturing Systems." *International Journal of Production Research* 46 (19): 5469–5490.

Schumacher, F., and A. Fay. 2014. "Formal Representation of GRAFCET to Automatically Generate Control Code." *Control Engineering Practice* 33: 84–93.

Seow, K. T. 2014. "Organizational Control of Discrete-event Systems: A Hierarchical Multi-world Supervisor Design". *IEEE Transactions on Control Systems Technology* 29 (2): 695–1306.

Shu, S., and F. Lin. 2014. "Decentralized Control of Networked Discrete Event Systems with Communication Delays." *Automatica* 50: 2108–2112.

Silva, D. B., A. D. Vieira, E. F. R. Loures, M. A. Busetti, and E. A. P. Santos. 2011. "Dealing with Routing in an Automated Manufacturing Cell: A Supervisory Control Theory Application." *International Journal of Production Research* 49: 4979–4998.

Tajer, A., A. Philippot, and V. Carré-Ménétrier. 2011. "Distributed Optimal Controller for Manufacturing Systems through Synthesis Approach." *The IEEE International Conference on Communication, Computing and Control Applications (CCCA'11)*. Hammamet.

Tajer, A., A. Philippot, and V. Carré-Ménétrier. 2013. "Centralised Controller for Manufacturing Systems through Liveness Extraction Approach." *International Journal of Systems, Control and Communications* 5 (3): 189–213.

W3C. 2012. "Extensible Markup Language (XML) 1.0". *World Wide Web Consortium* 5th ed. Accessed 5, 2012. http://www.w3.org/TR/xml/

Wang, L., C. Mahulea, and M. Silva. 2013. "Distributed Model Predictive Control of Timed Continuous Petri Nets." In *IEEE 52nd Annual Conference on Decision and Control (CDC)*, 6317–6322. IEEE. Florence.

Wong, K. C., and W. M. Wonham. 1996. "Hierarchical Control of Discrete-event Systems." *Discrete Event Dynamic Systems* 6 (3): 241–273.

Wonham, W.M. 2012. *Supervisory Control of Discrete-event Systems*. Systems Control Group, Department of Electrical and Computer Engineering, University of Toronto. http://www.control.toronto.edu/DES.

Wonham, W. M., and J. G. Ramadge. 1988. "Modular Supervisory Control of Discrete Event Systems." *Mathematics of Control, Signals, and Systems* 1 (1): 13–30.

Yang, Y., H. Hu, and Y. Liu. 2015. "A Petri Net-based Distributed Control of Automated Manufacturing Systems with Assembly Operations." In *2015 IEEE International Conference on Automation Science and Engineering (CASE)*, 1090–1097. Gothenburg.

Zaytoon, J., and V. Carré-Ménétrier. 2001. "Synthesis of Control Implementation for Discrete Manufacturing Systems." *International Journal of Production Research* 39 (2): 329–345.


écoles .urca
Doctorales