# Gestion des modes de systèmes à événements discrets : application au passage de frontière sous ERTMS

Hela Kadri

**N° d'ordre : 391**

**Centrale Lille**

# THÈSE

présentée en vue d'obtenir le grade de

# DOCTEUR

en

*Spécialité : Automatique, Génie Informatique, Traitement du Signal et Images*

par

**Hela KADRI ép DKHIL**

**DOCTORAT délivré conjointement par CENTRALE LILLE et FACULTÉ DES SCIENCES DE TUNIS**

Titre de la thèse :

## Gestion des modes de systèmes à événements discrets : application au passage de frontière sous ERTMS

Soutenue le 20 Janvier 2020 devant le jury d'examen :

| | | |
|---|---|---|
| **Président** | Pr, Kamel BARKAOUI | CNAM, Paris |
| **Rapporteur** | MCF-HDR, Audine SUBIAS | LAAS-CNRS, Toulouse |
| **Rapporteur** | Pr, Moncef TAGINA | ENSI, Tunis |
| **Membre** | Pr, Khaled GHEDIRA | Université Centrale, Tunis |
| **Membre** | Pr, Leila BEN AYED | ENSI, Tunis |
| **Membre** | CR, Christophe GRANSART | Univ Gustave Eiffel, Lille |
| **Directeur de thèse** | DR1, Simon COLLART-DUTILLEUL | Univ Gustave Eiffel, Lille |
| **Directeur de thèse** | Pr, Samir BEN AHMED | FST, Tunis |
| **Encadrant** | CR, Philippe BON | Univ Gustave Eiffel, Lille |

Thèse préparée dans le Laboratoire d'Évaluation des Systèmes de Transports Automatisés et de leur Sécurité COSYS-ESTAS,
Univ Gustave Eiffel, IFSTTAR, Univ Lille, F-59650 Villeneuve d'Ascq, France

École Doctorale SPI 072 (EC Lille)

# Remerciements

Cette thèse s'inscrit dans le cadre d'une thèse en cotutelle entre Centrale Lille et la Faculté des Sciences de Tunis (FST). Elle s'est effectuée sous la direction de Monsieur *Simon Collart-Dutilleul*, Directeur de Recherche à l'Université Gustave Eiffel, et Monsieur *Philippe Bon*, Chargé de Recherche à l'Université Gustave Eiffel pour la part française et sous la direction de Monsieur *Samir Ben Ahmed*, Professeur à la FST pour la part tunisinne. Ces travaux ont été réalisés en grande partie au laboratoire Évaluation des Systèmes de Transports Automatisés et de leur Sécurité (ESTAS) du département COSYS de l'Université Gustave Eiffel-Villeneuve d'Ascq.

C'est avec toute ma profonde gratitude que j'exprime mes sincères remerciements à Monsieur *Simon Collart-Dutilleul* pour ses hautes qualités humaines, ses conseils, ses discussions fructueuses. Sans doute, ce travail n'aurait pas pu être mené à son terme sans son soutien attentif et la patience qu'il m'a accordée.

Je suis extrêmement reconnaissante à Monsieur *Samir Ben Ahmed*, qui m'a fait le grand honneur de bien vouloir m'encadrer dans ce travail, pour ses précieuses remarques. Qu'il trouve ici l'expression de ma profonde gratitude pour toute l'attention, le soutien et le temps qu'il m'a consacré.

Mes remerciements les plus chaleureux s'adressent à Monsieur *Philippe Bon* pour l'encadrement de ce travail de thèse, pour son soutien, ses conseils judicieux et la disponibilité qu'il a témoignée à mon égard.

Mes vifs remerciements vont aussi à Madame *Audine Subias*, Maître de conférence HDR au LAAS-CNRS DISCO Team de Toulouse et Monsieur *Moncef Tagina*, Professeur à l'École Nationale des Sciences de l'Informatique de Manouba pour avoir accepté d'examiner ce travail en qualité de rapporteurs. Je tiens également à exprimer ma gratitude à Madame *Leila Ben Ayed*, Professeur à l'École Nationale des Sciences de l'Informatique, et Messieurs *Kamel Barkaoui*, Professeur au Conservatoire National des Arts et Métiers, et *Khaled Ghedira*, Recteur du Groupe Université Centrale, pour avoir accepté de faire partie du jury en qualité d'examinateurs, ainsi que Monsieur *Christophe Gransart*, Chargé de Recherche à l'Université Gustave Eiffel, en qualité d'invité.

Je tiens à remercier vivement Madame *Najiba Bellaaj*, Professeure à l'Institut Supérieur d'Informatique, pour ses conseils judicieux et pour tous les bons moments passés ensemble.

Merci pour votre confiance et votre fidélité.

Je tiens vraiment à remercier *Abderraouf Boussif* pour tout l'aide et le support moral qu'il m'a accordé pendant cette thèse. Tu as agi comme un frère et je te remercie.

Mes remerciements vont également à tous les personnels de l'Université Gustave Eiffel-Villeneuve d'Ascq pour la gentillesse et la convivialité dont ils ont fait preuve et qui ont rendu mon séjour très agréable parmi eux ainsi que pour leur soutien et leurs encouragements.

Mes derniers remerciements iront évidemment à ma famille. Je pense tout d'abord à mes parents *Dalila* et *Farhat*, à ma soeur *Alya* et à mes deux frères *Issam* et *Wajdi* pour leur soutien et leur amour. Tout ça n'aurait jamais été possible sans leur soutien inconditionnel et leurs encouragements dans les moments difficiles. Merci pour avoir cru en moi et d'avoir fait de moi ce que je suis à présent.

Ensuite, je voudrais remercier mon cher mari *Fathi Dkhil*, qui partage ma vie et mes rêves, pour sa présence, sa confiance, son soutien, sa patience, et bien entendu pour son amour...

Enfin, Je remercie mes chers fils *Mohamed Amine* et *Youssef* dont l'amour est un carburant infini d'avancement et de dépassement de soi et ma chère fille *Lina Zeineb*, dont la naissance au cours de cette thèse m'a apporté beaucoup de joie.

# Publications

**Articles publiés**

— H. Kadri, et S. Collart Dutilleul. Multi-Objective Optimization for Path Searching in a Flow Network with Maintenance Tasks. 14th International Conference on Control, Automation, Robotics & Vision (ICARCV 2016), Phuket, Thailand, 13-15th November 2016.

— H. Kadri, S. Ben Ahmed, et S. Collart Dutilleul. Formal Approach to Control Design of Complex and Dynamical Systems. Journal of Procedia Computer Science 108C, pp.2512–2516, 2017.

— H. Kadri, S. Schleiner, S. Collart Dutilleul, P. Bon, S. Ben Ahmed, F. Steyer, A. Gabriel, O. A. Mudimu. Proposition of a formal model for crisis management in the context of high-speed train networks in border areas. 7th Transport Research Arena, TRA 2018, Vienne, Austria, 2018.

— H. Kadri, S. Collart Dutilleul, P. Bon, et S. Ben Ahmed. A Formal Approach for Multi-occurrence Crisis Management. 13th International Conference on Software Technologies, ICSOFT 2018, Porto, Portugal, 2018.

— H. Kadri, S. Collart Dutilleul, P. Bon, et S. Ben Ahmed. Formal Approach to Dynamic SoS Design. 14th International Conference on Evaluation of Novel Approaches to Software Engineering, ENASE2019, HERAKLION, CRETE - GREECE, 2019.

**Articles acceptés**

— S. Collart Dutilleul, H. Kadri, P. Bon, G. Mykoniatisb, et S. Ben Ahmed. Security and safety integrated approach for multimodal-hubs crisis management : a railway and airway proposition. 8th Transport Research Arena, TRA 2020, Helsinki, Finland, 2020.

**Articles soumis**

— H. Kadri, S. Collart Dutilleul, P. Bon, et S. Ben Ahmed. Colored Petri Net Model for Control Problem of Border Crossing Under Constraints. Journal of Universal Computer Science (JUCS), 2018.

# Table des matières

# Table des figures

# Liste des tableaux

# Liste des définitions

# Liste des Algorithmes

# Acronymes

**BSI** British Standards Institution.

**EGC** Equipe de Gestion de Crise.
**ERTMS** European Rail Traffic Management System.
**ETCS** European Train Control System.

**FS** Full Supervision.

**GSM-R** Global System for Mobile Communications – Railway.

**IC** Infrastrure Critique.
**IS** Isolation.

**LEU** Lineside Electronic Unit.
**LS** Limited Supervision.

**MA** Movement Authority.

**NL** No leading.
**NP** No Power.

**OS** On Sight.

**PGC** Plan de Gestion de Crise.
**PT** Post Trip.

**RBC** Radio block center.
**RdPC** Réseaux de Petri colorés.
**RdPCH** Réseaux de Petri colorés hiérarchiques.
**RdPCP** Réseau de Petri Coloré à Priorité.
**RdPHN** Réseaux de Petri de Haut Niveau.
**RV** Reversing.

**SAP** Systèmes Automatisés de Production.
**SB** Stand-By.
**SdS** Système-de-Systèmes.

**SED** Système à Evènnement Discret.

**SF** System Failure.

**SGC** Système de Gestion de Crise.

**SGCP** Système de Gestion de Crise d'un Pays.

**SH** Shunting.

**SIGC** Système Internationale de Gestion de Crise.

**SL** Sleeping.

**SN** STM National.

**SR** Staff Responsible.

**TCS** Théorie de Contrôle par Supervision.

**TR** Trip.

**UE** Union Européen.

**UIC** Union Internationale des Chemins de fer.

**UN** Unfitted.

# Chapitre 1

# Introduction

L'interopérabilité ferroviaire a pour but de créer un réseau ferroviaire permettant un transport sûr, conforme au niveau de performance requis des lignes, et ne nécessitant pas de changement de trains. Ainsi le respect d'un ensemble de règles, de conditions techniques et opérationnelles garantissant le respect des exigences essentielles est nécessaire. La proposition européenne correspondante consiste à mettre en place un système européen de gestion du trafic ferroviaire appelé ERTMS «European Rail Traffic Management System». La gestion de la signalisation ferroviaire sous ce système repose sur des règles de signalisation propres à chaque pays et non sur des règles globales. La spécification ERTMS est plutôt de haut niveau, par conséquent un besoin de raffinement, au sens de l'ingénierie logicielle, est nécessaire pour faire le lien entre les règles nationales et la spécification ERTMS. Les possibilités de raffinement sont effectives, mais le choix doit être fait en intégrant les exigences de sécurité globales. En effet, entre les règles de bord et celles du sol, il existe des règles d'exploitation qui définissent des modalités spécifiques d'application du système ERTMS sur une infrastructure ferroviaire donnée. Dans ces règles d'exploitation, certains modes opératoires globaux ERTMS sont parfois interdits. Cette dernière remarque apporte une difficulté à franchir une frontière sans reconfigurations lourdes. Au minimum, nous souhaitons qu'un train entre dans un nouveau pays sans s'arrêter, en utilisant le système ERTMS.

Dans ce cadre, cette thèse vise à définir des moyens pour effectuer un passage de frontière en sécurité. Concrètement, nous conserverons un ensemble de propriétés de sécurité dans les zones frontalières permettant le franchissement de la frontière. Pour ce problème, le système ERTMS est étudié comme un Système-de-Systèmes (SdS) dans la mesure où il se compose essentiellement de deux systèmes à la fois indépendants, autonomes et complexes : le système à bord, embarqué dans les trains, et le système sol, équipé aux infrastructures, communiquant via une radio GSM-R (Global System for Mobile Communications – Railway). De plus, lors d'un franchissement de frontière, les trains changent d'interlocuteur

c'est-à-dire de système sol. Notre intérêt réside dans le fait que le concept de SdS permet d'avoir une vue globale du système de signalisation en tenant compte de la diversité des systèmes qui le constituent ainsi que des différentes relations et interactions entre eux. Par ailleurs, nous avons analysé les résultats et les théories qui permettent de décrire les systèmes critiques notamment les Systèmes à Événements Discrets (SEDs) tout en se focalisant sur la maîtrise de leur comportement. Parmi les méthodes fournissant des réponses significatives à ce contexte, la Théorie de Contrôle par Supervision (TCS) se distingue en offrant un cadre formel pour prendre en compte a priori les propriétés attendues du système pour synthétiser l'ensemble des comportements admissibles [Ramadge et Wonham 1987]. De plus, une des notions couramment utilisée pour la conception de la commande des SEDs est la notion de mode de fonctionnement car elle diminue considérablement la complexité des modèles manipulés par la décomposition. Il existe peu sont les méthodes qui ont été spécifiquement conçues pour appréhender la gestion des modes de fonctionnement, néanmoins une approche intéressante dite multi-modèle a été développé dans le cadre des SEDs et particulièrement les Systèmes Automatisés de Production (SAP) [Kamach 2004] qui sont des systèmes dynamiques et susceptible d'évoluer entre des modes de fonctionnement normaux et défaillants sans arrêter. L'approche multi-modèle consiste à intégrer la gestion des modes de fonctionnement dans la TCS et à contrôler la commutation entre les modes tout en maintenant un fonctionnement acceptable du système, suite à l'occurrence d'événements.

Ainsi, et sur la base de ce qui précède, l'objectif de cette thèse est de proposer une méthodologie de conception sûre du comportement dynamique des SdSs. Premièrement, nous proposons une démarche sûre de conception des systèmes complexes basée sur la gestion des modes de fonctionnement. Deuxièmement, nous présentons une généralisation de la démarche pour couvrir les SdSs en tenant compte des dépendances inter-systèmes lors des défaillances. Troisièmement, nous considérons ERTMS Niveau 2 comme un SdS et nous cherchons à résoudre le problème de franchissement de la frontière. Quatrièmement, dans une optique de généralisation de nos propositions, nous appliquons nos résultats aux systèmes de gestion de crises.

Ce mémoire est organisé comme suit. Le chapitre 1 présente le contexte industriel et les bases théoriques du présent travail. Il commence par présenter le système ERTMS et décrire la problématique scientifique issue de la diversité technique et infrastructurelle de réseau ferroviaire européen. Soutenu par une revue des travaux, il énonce ensuite les fondements théoriques sur lequel reposent nos propositions. Le chapitre 2 développe une démarche de conception des systèmes dynamiques en adoptant une décomposition modale. Le chapitre 3 présente une généralisation qui s'applique aux SdSs en se concentrant sur les dépendances systèmes. Le chapitre 4 propose une approche modale basé sur le concept de SdS pour résoudre le problème du franchissement des frontières en utilisant ERTMS.

Le chapitre 5 est une application de nos résultats dans le domaine de la gestion de crise. Enfin, le chapitre 6 présente les conclusions et perspectives de ce travail.

# Chapitre 2

# Problématique et état de l'art

## 2.1 Introduction

Le transport ferroviaire est devenu un moyen de transport massivement utilisé notamment en Europe (transport de marchandises, transport interurbain, intra-urbain et transport à grande vitesse). L'interopérabilité des systèmes ferroviaires est très importante car elle permet la circulation sûre et sans rupture des trains, tout en accomplissant les performances spécifiées et en respectant les contraintes techniques et légales spécifiques à chaque pays. L'amélioration de la compatibilité des différents systèmes via l'harmonisation technique est un objectif important du domaine ferroviaire. Au niveau européen, un consortium appelé EUROSIG a été fondé et il a lancé un système de signalisation standard appelé ERTMS visant à assurer l'interopérabilité des trains. Cependant, le franchissement de frontière reste encore un des obstacles techniques.
Ce chapitre se compose de deux sections principales. Dans la première section, nous introduisons le système ERTMS et nous développons la problématique scientifique lié au franchissement de la frontière en utilisant le système ERTMS Niveau 2. Dans la deuxième section, nous présentons la perspective scientifique avec laquelle est étudié le système ERTMS, à savoir les SEDs et les SdSs. Nous présentons également la théorie des Réseaux de Petri de Haut Niveau (RdPHNs) et la TCS qui définissent le cadre de base sur lequel reposent nos propositions.

## 2.2 Contexte industriel

### 2.2.1 Aperçu sur l'ERTMS/ETCS

Conçu à l'origine pour remplacer les différents systèmes nationaux des pays de l'UE, ERTMS est devenu la principale norme internationale pour les systèmes de contrôle et

de commande des trains [ECDGET 2006, Guiot et al. 2009] et il est devenu une référence mondiale utilisée dans plusieurs pays comme le Brésil, la Mexique, l'Australie et la Chine.

Une composante principale du système ERTMS est le Système européen de contrôle des trains ETCS (European Train Control System), qui est un système de signalisation de cabine et de contrôle de vitesse. En raison de la nature des fonctions requises, le système ERTMS/ETCS est en partie sur le sol et en partie à bord des trains. Ceci définit deux systèmes, le système «sol» et le système «bord».

#### Le système sol

Dépendant du niveau d'application (voir sous-section 2.2.2 suivante), le système sol peut être composé de plusieurs composants détaillés ci-dessous :

**Balises :** ce sont des dispositifs de transmission appelés aussi «Eurobalises®». Ils peuvent envoyer des télégrammes du sol au système bord.

**Codeur ERTMS (LEU) :** (Lineside Electronic Unit) il génère des télégrammes à envoyer par les balises, sur la base des informations reçues par systèmes de sol externes (interlockings, les centres de contrôle, etc.).

**GSM-R :** c'est une norme internationale de communication téléphonique pour mobile dédié aux chemins de fer, il est utilisé pour l'échange bidirectionnel des messages entre des systèmes sol et les RBCs.

**Radio Block Center (RBC) :** c'est un système informatisé qui élabore des messages à envoyer au train sur la base des informations reçues des systèmes de sol externes et sur la base des informations échangées avec les systèmes bord.

#### Le système bord

Dépendant aussi du niveau d'application (voir sous section 2.2.2 suivante), le système bord peut être composé des éléments suivants :

**Equipement ERTMS/ETCS à bord :** c'est un système informatisé qui supervise le mouvement du train auquel il appartient, sur la base des informations échangées avec le système sol.

**GSM-R à bord de radiocommunication :** c'est la norme utilisée pour l'échange bidirectionnel de messages entre un système bord et le RBC.

### 2.2.2 Niveaux ERTMS

Les niveaux ERTMS définissent les différentes utilisations d'ERTMS en tant que système de contrôle de train, s'étendant des communications sol au train (Niveau 1) aux

communications continues entre le train et le RBC (Niveau 2). Le Niveau 3, qui est encore dans une phase conceptuelle, augmentera encore le potentiel du système en introduisant une technologie «moving block».

### ERTMS Niveau 1

ERTMS Niveau 1 est conçu comme une superposition d'une ligne conventionnelle, déjà équipée de signaux de sol, et de détecteurs de train. La communication entre le sol et le train est assurée par des balises situées sur le bord de la piste adjacente aux signaux de sol à intervalles réguliers et reliées au centre de contrôle des trains.
En recevant le message Movement Authority (MA) via les balises, l'équipement embarqué ETCS calcule automatiquement la vitesse maximale du train et le prochain point de freinage si nécessaire, en prenant en considération les caractéristiques de freinage du train et les données de description de la voie. Cette information est affichée au conducteur à travers un écran dédié dans la cabine. La vitesse du train est surveillée en permanence par l'équipement embarqué ETCS.
Les principaux avantages apportés par ERTMS Niveau 1 sont l'interopérabilité (entre les fournisseurs et les pays) et la sécurité, puisque le train freine automatiquement s'il dépasse la vitesse maximale autorisée par le MA.

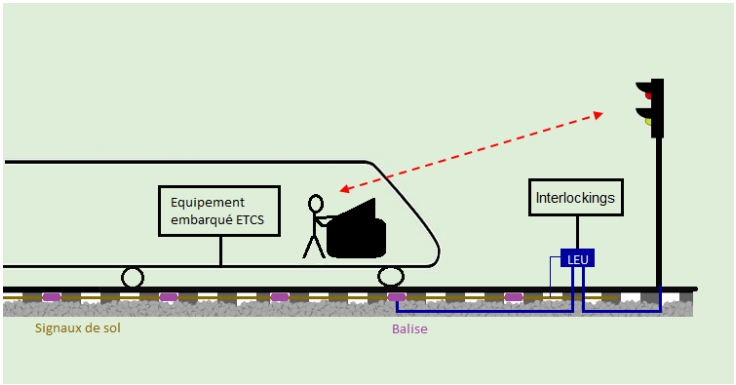



Figure 2.1 – Architecture d'un ERTMS/ETCS Niveau 1

### ERTMS Niveau 2

Contrairement au Niveau 1, le Niveau 2 n'exige pas de signaux côté sol. Le MA est communiquée directement du RBC à l'équipement ERTMS/ETCS à bord via le GSM-R. Les balises ne sont utilisées que pour transmettre des «messages fixes» tels que la position, le gradient, la limitation de vitesse, etc. Un flux continu de données informe le conducteur des données spécifiques à la ligne et de l'état des signaux de sol à suivre, permettant ainsi au train d'atteindre sa vitesse maximale ou optimale tout en maintenant une distance de

freinage sûre, et tout en permettant des coûts de construction et de maintenance fortement réduits par la suppression des signaux côté sol. ERTMS Niveau 2 offre également la possibilité d'augmenter considérablement la capacité de ligne en permettant des vitesses opérationnelles plus élevées et en réduisant généralement les intervalles entre les passages des trains : plus de capacité signifie plus de trains en mouvement, donc une meilleure rentabilité de l'infrastructure et un meilleur niveau de service.
Le Niveau 2 de l'ERTMS offre les mêmes avantages en matière d'interopérabilité et de sécurité que le Niveau 1.

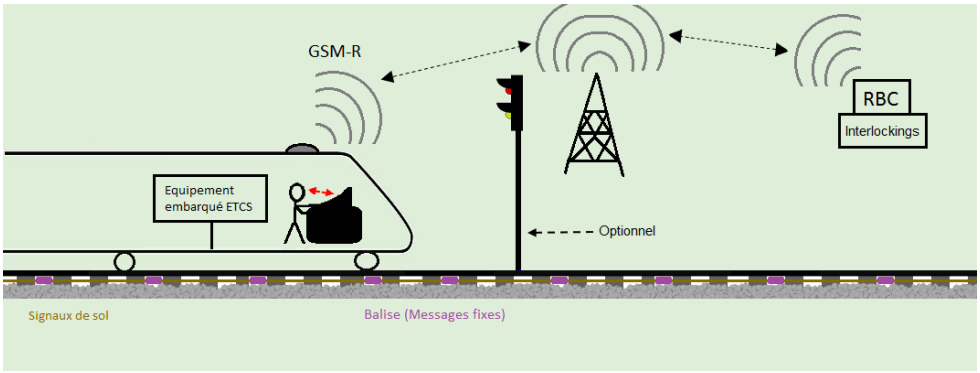



Figure 2.2 – Architecture d'un ERTMS/ETCS Niveau 2

**ERTMS Niveau 3**

ERTMS Niveau 3 est toujours dans sa phase de conception et il introduit une nouvelle technologie appelé «bloc mobile». Dans les niveaux 1 et 2 du système ERTMS, les MA sont déterminées à l'aide de «blocs fixes» : sections de voie entre deux points fixes qui ne peuvent pas être utilisés simultanément par deux trains. Avec ERTMS Niveau 3, les données de position précises et continues sont fournies au centre de contrôle directement par le train plutôt que par l'équipement de détection basé sur le sol. Comme le train surveille en permanence sa propre position, il n'est pas nécessaire de disposer de «blocs fixes» ; le train lui-même sera considéré comme un bloc mobile

### 2.2.3 Modes opératoires sous ERTMS

ERTMS présente différents modes opératoires permettant de superviser totalement ou partiellement les déplacements des trains en fonction des conditions d'exploitation de la ligne ou de l'état de l'équipement embarqué. Chacun de ces modes garantit un niveau de sécurité de conduite différent. Ci-dessous une liste des modes d'ERTMS [UNISIG 2008] :

**Full Supervision (FS) :** c'est le mode nominal de mouvement où le train est autorisé à rouler à la vitesse maximale indiquée et il est supervisé par rapport à un profil de

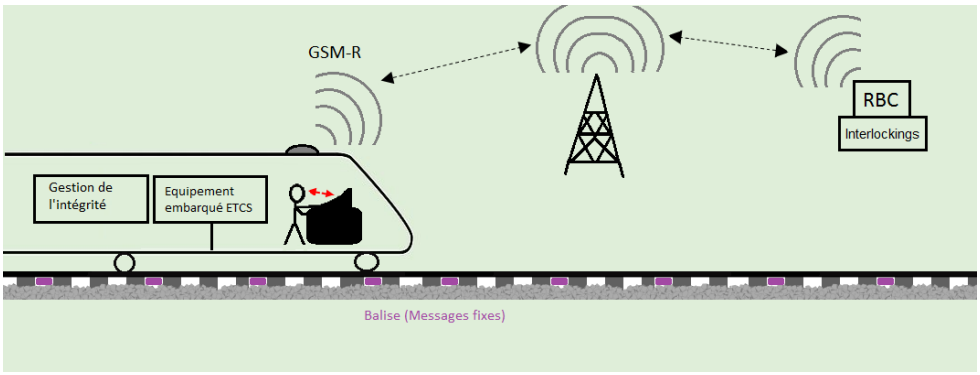



Figure 2.3 – Architecture d'un ERTMS/ETCS Niveau 3

vitesse dynamique. Ce mode ne peut pas être sélectionné par le conducteur, mais il est automatiquement enclenché par l'unité embarquée lorsque toutes les données de train et de piste, nécessaires à une supervision complète, sont disponibles à bord.

**Isolation (IS) :** ce mode s'applique lorsque le conducteur est isolé du système ERTMS suite à une panne.

**No Power (NP) :** ce mode est activé lorsque l'équipement embarqué ERTMS/ ETCS n'est pas sous tension. Il est accompagné d'une demande de freinage d'urgence.

**On Sight (OS) :** ce mode permet au train d'entrer dans une section de voie qui pourrait déjà être occupée par un autre train ou être obstruée par un autre type d'obstacle. Le train doit avancer à une vitesse maximale, qui est paramétrable nationalement, suffisante pour pouvoir s'arrêter avant l'obstacle.

**Staff Responsible (SR) :** Ce mode permet de déplacer le train sous la responsabilité du conducteur sur une ligne équipée du système ERTMS. La vitesse du train est supervisée par une vitesse maximale paramétrable nationalement.

**Shunting (SH) :** ce mode est utilisé lors des mouvements de manoeuvre. Le train se déplace sans que des données, comme la position, ne lui soient parvenues. La vitesse du train est affichée et elle est supervisée par une vitesse maximale. Ce mode peut être activé par le pilote ou commandé par voie.

**Stand-By(SB) :** c'est le mode par défaut, il est automatiquement sélectionné lors de l'ouverture ou de la fermeture du poste de conduite. L'équipement embarqué ERTMS/ETCS assure la surveillance à l'arrêt. Il correspond à un mode d'attente.

**Sleeping (SL) :** ce mode est défini pour gérer l'équipement embarqué ERTMS/ ETCS d'une locomotive esclave contrôlé à distance.

**Unfitted (UN) :** Ce mode est sélectionné lorsque un train équipé du système ERTMS traverse des lignes non équipées ERTMS. La vitesse du train est uniquement affichée au conducteur et elle est supervisée jusqu'à une vitesse maximale paramétrable nationalement.

**No leading (NL) :** ce mode est défini pour gérer l'équipement embarqué ERTMS/ETCS d'une locomotive esclave qui n'est pas couplée électriquement à une locomotive «maîtresse» (et donc n'est pas télécommandée) mais qui possède son propre pilote.

**Trip (TR) :** ce mode est automatiquement sélectionné en cas de dépassement de MA. Une demande de freinage d'urgence se produira. Un accusé de réception du freinage d'urgence sera lancé.

**Post Trip (PT) :** ce mode est activé automatiquement, après un arrêt d'urgence du train en mode Trip, une fois que le train est arrivé à un état d'arrêt et que le conducteur a accusé réception du mode Trip.

**System Failure (SF) :** ce mode est associé à une défaillance de l'équipement embarqué ERTMS/ETCS et est accompagné d'une demande de freinage d'urgence.

**STM National (SN) :** ce mode permet au système national d'accéder à certaines ressources de l'équipement embarqué ERTMS/ETCS.

**Reversing (RV) :** ce mode permet au conducteur de changer le sens du mouvement du train et de partir depuis la même cabine, c'est-à-dire rouler en marche arrière. Ce mode est utilisé pour permettre au train de sortir d'une situation dangereuse et d'atteindre le plus rapidement possible une situation «plus sûre». Dans ce mode, la vitesse maximale et la distance autorisée sont supervisées.

**Limited Supervision (LS) :** ce mode permet le fonctionnement du train dans des zones où les informations au sol peuvent être fournies afin de réaliser une supervision en arrière-plan du train. Ce mode ne peut pas être sélectionnée par le conducteur, mais il doit être enclenché automatiquement lorsque système bord reçoit les commandes et que toutes les conditions nécessaires sont remplies. Le conducteur doit alors respecter la signalisation extérieure, puisque toutes les informations ne sont pas remontées à bord.

### 2.2.4 Problématique scientifique

Le réseau ferroviaire de l'UE se caractérise par sa grande taille et sa diversité technique et infrastructurelle où plusieurs trains internationaux opèrent ; comme par exemple Intercity-Express, qui est un système de trains à grande vitesse circulant principalement en France, en Allemagne, en Autriche, en Suisse, au Danemark et dans les pays voisins ; Thalys, qui est un opérateur de train à grande vitesse franco-belge circulant en France, en Allemagne, en Belgique et en Pays-Bas ; et le TGV, qui est le service ferroviaire interurbain à grande vitesse en France, exploité par la SNCF, l'opérateur ferroviaire national ; ses services s'étendent à la Belgique, à l'Italie, en Suisse, en Espagne, en Allemagne et au Luxembourg.

Cependant, et malgré le nombre important de trains qui traversent les frontières de l'UE, le franchissement de la frontière en utilisant ERTMS est peu usité. Cela vient du fait que la construction du réseau ferroviaire européen repose sur la connexion des réseaux ferroviaires nationaux. Par conséquent, le comportement du système sol, qui en résulte, est spécifique à chaque pays et n'est d'ailleurs pas décrit dans la spécification ERTMS.
L'une des conséquences est qu'à un passage frontalier, le système sol change et que certains modes opératoires ERTMS interdits dans un pays deviennent autorisés dans un autre et les vitesses maximales autorisées, des modes opératoires communs, peuvent également changer en fonction des contraintes spécifiques à chaque pays.
Ce problème frontalier est le point de départ de nos travaux de recherche. Nous pouvons l'exprimer de la manière suivante : soit un ensemble de trains, équipés chacun d'un système bord, qui opèrent sur des infrastructures nationales équipées chacune d'un système sol spécifique. Chaque train interagit avec un ou plusieurs systèmes sol et chaque système sol interagit à son tour avec un ou plusieurs trains. Pour assurer le passage frontalier des trains sous ERTMS Niveau 2, nous avons pensé à l'ajout des modes transitoires dont le rôle est de préparer les trains à franchir la frontière. Ainsi, au niveau de chaque pays, le système sol sera décomposé en un mode nominal, représentant son fonctionnement normal, et en un ensemble de modes transitoires. Les trains, en fonction de leurs positions, commutent de modes et de systèmes sol. Cette décomposition modale nous évite d'étudier le problème dans sa globalité puisque chaque mode ne s'intéresse qu'à une période de fonctionnement et à un ensemble de composants plus réduits. Pour se focaliser sur la maîtrise du comportement, nous appliquerons la gestion des modes de fonctionnement qui vise à maintenir un fonctionnement acceptable du système lors des commutations tout en s'appuyant sur la TCS. Nous adopterons aussi l'approche multi-modèle, dans laquelle un modèle différent est associé à chaque mode de fonctionnement. Ceci permet de définir une stratégie de contrôle appropriée pour chaque mode de fonctionnement.
Par ailleurs, dans le but de fournir une vue globale du système ferroviaire européen ainsi décomposé, nous présenterons une modélisation basée sur le concept de SdS. La section suivante est consacrée à la présentation du cadre théorique dans lequel nous positionnons notre problématique, à savoir les SdSs et la TCS. Nous présentons également le langage de modélisation adopté qui est les RdPHNs.

## 2.3 Base théorique

### 2.3.1 Système-de-systèmes

Ces dernières années ont vu se développer un intérêt croissant pour les systèmes complexes dont les constituants sont également complexes. La modélisation et la robustesse des

systèmes complexes dans lesquels des composants indépendants et hétérogènes coopèrent pour atteindre un objectif commun sont des tâches essentielles dans de nombreuses applications. Le concept de système-de-systèmes (SdS) a été proposé dans le contexte de ces systèmes complexes. C'est un concept offrant un point de vue de haut niveau englobant les interactions entre les systèmes indépendants coopérants [Jamshidi 2008]. [Nanayakkara et al. 2009] ont introduit un certain nombre de sujets fondamentaux sur les SdSs, notamment la définition, la modélisation, les applications et l'évaluation. Le concept SdS est appliquée dans divers domaines tels que les transports [DeLaurentis 2005], la santé [Wickramasinghe et al. 2008], l'environnement des navires [Mahulkar et al. 2009], la robotique [Jamshidi 2008], le militaire [Huynh et Osmundson 2006], etc.

### Définition

[ISO/IEC/IEEE 2015] définit un SdS comme un ensemble de systèmes réunis pour une tâche qu'aucun système ne peut accomplir seul. Chaque système constituant le SdS conserve sa propre gestion, ses objectifs et ses ressources tout en se coordonnant avec les autres au sein du SdS et en s'adaptant pour atteindre ses objectifs. Néanmoins, il existe plusieurs définitions. Avant la définition ci-dessus, plusieurs définitions ont été données mais elles ne sont pas universellement acceptées. De plus, certaines de ces définitions dépendent de la particularité d'un domaine d'application. Par exemple, [Checkland 1999] définit un SdS comme deux ou plusieurs systèmes définis séparément mais fonctionnant ensemble pour réaliser un objectif commun. Le ministère de la défense américain [DoD 2008] considère un SdS comme un ensemble ou une composition de systèmes résultant de l'intégration de systèmes utiles et indépendants dans un système plus vaste offrant des capacités uniques. [Maier 1998] propose cinq traits, connus sous le nom de critères de Maier, permettant de distinguer les SdSs : indépendance opérationnelle des éléments, indépendance de gestion des éléments, développement évolutif, comportement émergent, et répartition géographique des éléments. Le conseil international d'ingénierie des systèmes [INCOSE 2012] considère que le terme système-de-systèmes devrait s'appliquer à un système d'intérêt dont les éléments de système sont eux-mêmes des systèmes ; ceux-ci impliquent généralement des problèmes de discipline avec de multiples systèmes hétérogènes distribués.

### Modélisation des SdSs

Les SdSs sont un concept offrant un point de vue de haut niveau englobant les interactions entre les systèmes indépendants coopérants. Différents modèles ont été proposés dans la littérature pour les modéliser. [Huynh et Osmundson 2006] ont considéré le système de protection du domaine maritime comme un SdS et ils ont utilisé plusieurs diagrammes

SysML (Systems Modeling Language) pour le modéliser et le simuler. [Eusgeld et al. 2011] ont traité les infrastructures critiques comme un SdS et ils ont proposé deux alternatives de modélisation pour les SdSs : un modèle intégré contenant des modèles détaillés de systèmes de bas niveau, ainsi qu'un modèle de haut niveau et un modèle couplé qui regroupe les sorties des modèles de bas niveau en tant qu'entrées de niveau supérieur. [Qiu 2014] a considéré le système ERTMS comme un SdS et elle a proposé un modèle dysfonctionnel du SdS intégrant les aspects matériels, les aspects réseaux ainsi que le facteur humain à l'aide de Statecharts et VBS (Valuation-Based System). Elle a également analysé les incertitudes en utilisant la théorie de Dempster-Shafer.

### 2.3.2 Contrôle de supervision des SEDs

#### Systèmes à événements discrets

Les Systèmes à Événement Discret (SEDs) sont une abstraction pour une grande variété de problèmes [Fabian et Hellgren 1998]. Ils représentent tous les systèmes pouvant être exprimés dans un état discret. Par exemple, les systèmes embarqués, les systèmes de production, les systèmes de trafic (aérien, ferroviaire, etc.), les protocoles de communication, etc. Ce sont des systèmes dynamiques dont l'activité est due aux occurrences asynchrones d'événements discrets, dont certaines sont provoquées (appui sur une touche du clavier) et d'autres pas (panne spontanée d'un équipement). Dans un SED, l'espace d'état est un ensemble discret, ce qui signifie que le passage d'un état à un autre se produit de manière abrupte. De plus, la fonction de transition associe toujours l'occurrence d'un événement dans l'état discret actuel à un autre état discret c'est-à-dire que les changements d'état (appelés transitions) sont déclenchés par des événements [Cassandras et Lafortune 2008].

#### Théorie du contrôle de supervision

En 1987, Ramadge et Wonham ont proposé la Théorie du Contrôle de Supervision (TCS) pour la synthèse des contrôleurs sous la forme de superviseurs [Ramadge et Wonham 1987, Wonham et Ramadge 1988, Ramadge et Wonham 1989]. Elle assume que les systèmes à l'étude peuvent être représentés sous forme de DESs. La TCS est largement appliquée dans le contexte de la production industrielle qui à leur tour a fait du TCS une pratique fiable [Fabian et Hellgren 1998, Liu et Darabi 2002, Silva et Queiroz 2010, Leal et al. 2012, Pinheiro et al. 2015].
Dans la TCS, les langages formels sont utilisés pour modéliser les capacités des systèmes. Dans le même temps, des spécifications, également exprimées en langage formel, sont utilisées pour limiter ces capacités. Ceci garantit que le système se comporte comme prévu. Cette théorie se fonde sur les automates à états finis et les langages formels [Wonham

2000, Wonham 2003]. La TCS distingue les événements qui déterminent l'évolution du système entre événements incontrôlables (représentent une entrée de contrôle) et événements contrôlables (représentent la sortie de contrôle). Dans la TCS, le concepteur modélise (i) ce que le système peut faire et (ii) ce qu'il devrait faire. Concernant (i), ils spécifient un nombre arbitraire de modèles de comportement dit libre, décrivant toutes les capacités du système. Concernant (ii), ils spécifient un nombre arbitraire de prétendues spécifications de contrôle. Les modèles de comportement libre et les spécifications de contrôle sont exprimés à l'aide d'un langage formel. Chaque symbole de l'alphabet du langage correspond à un événement du SED. Par conséquent, la séquence souhaitable des événements forme les mots du langage. La TCS combine tous les modèles de comportement libre et les spécifications de contrôle dans un langage cohérent. Il synthétise un superviseur (contrôleur), qui garantit qu'à tout moment, seuls des mots valides ou des préfixes de mots valides apparaissent. Ceci est réalisé en limitant le nombre d'événements contrôlables que le système (procédé) peut choisir à travers une boucle de contrôle (voir Figure 2.4).

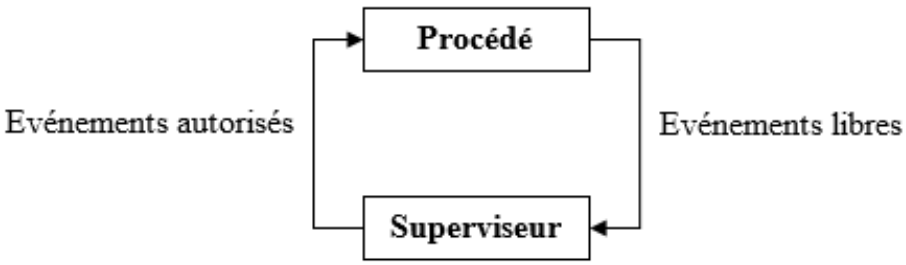



Figure 2.4 – Boucle de contrôle de supervision

Dans le cadre de la TCS, les mots d'un langage régulier, sont produits par le procédé, qui est similaire à un automate fini. Cependant, un automate fini reconnaît les mots d'un langage régulier donné (c'est-à-dire que l'automate l'acceptera ou ne l'acceptera pas), alors que le procédé produit des mots qui appartiennent au langage.

**Définition 1 (Automate)**
*Un automate $G$ est un 5-tuple :*

$$G =< Q, \Sigma, \delta, q_0, Q_m >$$

*où :*

$Q$ *est un ensemble fini d'états ;*

$\sigma$ *est un ensemble fini de symboles liés aux événements du système ;*

$\delta : Q \times \Sigma \rightarrow Q$ *est une fonction de transition partielle ;*

$q_0 \in Q$ *est l'état initial ;*

$Q_m \subseteq Q$ *est un ensemble d'états marqués.*

Le procédé est modélisé par un automate G qui génère un langage appelé $L(G)$. L'objectif de la TCS est de contrôler ce procédé par interdiction de certains comportements possibles du système étudié. Ceci revient à interdire des occurrences d'événements. Or, les événements - qui sont les symboles du langage - sont de deux types : les événements incontrôlables ($\Sigma_u$) et les événements contrôlables ($\Sigma_c$) où $\Sigma = \Sigma_u \cup \Sigma_c$ et $\Sigma_u \cap \Sigma_c = \emptyset$. Une panne, une information provenant d'un capteur, une date d'échéance, etc. sont des exemples d'événements incontrôlables.

Le rôle du superviseur S, associé au procédé, est donc d'interdire les occurrences d'événements non admissibles, et bien sûr il ne peut interdire que des événements contrôlables. Le superviseur est une fonction définie par $S : L(G) \rightarrow 2^{\Sigma}$. Ainsi, pour chaque mot $m$ généré par $G$, le superviseur $S$ lui renvoie l'ensemble des événements autorisés $S(m)$. $G$ ne peut alors évoluer qu'en générant un événement inclus dans cet ensemble.

### Approches de décomposition

**Limites de la théorie de Ramadge et Wonham** Le problème majeur limitant l'applicabilité de cette théorie est la taille du modèle de procédé. Par exemple, la modélisation de la ligne de transfert de l'AIP de Lyon a abouti à un procédé $G$ de 108000 états et 1165500 transitions [Pietrac et al. 2002] . Sur cette exemple, la synthèse de contrôleur n'a pas abouti car les calculs nécessaires dépassent les capacités du logiciel de synthèse «TCT» [1]. De même, il ne faut pas sous-estimer la difficulté de l'étape de modélisation et d'interprétation des modèles. Les approches suivantes cherchent à résoudre ce problème.

**Approche modulaire** L'approche modulaire, illustrée par la Figure 2.5, considère $n$ superviseurs ($n > 1$) pour un seul procédé. Chaque superviseur représente une spécification unique, observe tous les événements générés par le procédé et agit simultanément pour restreindre le comportement du procédé [Komenda et al. 2008]. Ainsi, un événement contrôlable n'est autorisé que s'il est autorisé par tous les superviseurs.

**Approche décentralisée** L'approche décentralisée reprend en partie l'approche modulaire dans la décomposition en plusieurs superviseurs de petites tailles et propose en plus la décomposition du procédé $G$ en plusieurs procédés locaux. À chaque procédé local est associé un superviseur local [Rudie et Wonham 1992]. La Figure 2.6 présente un exemple de décomposition du procédé global $G$ en deux procédés locaux $\{i = 1, 2\}$. La communi-

1. http ://www.control.utoronto.ca/people/profs/wonham/

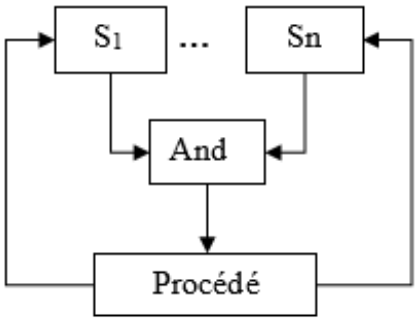



Figure 2.5 – Approche modulaire

cation est assurée par deux canaux d'information $P1$ et $P2$. Ceci peut être généralisé en un nombre quelconque de procédés locaux.

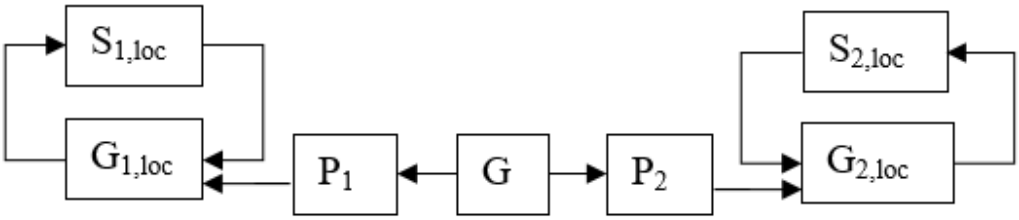



Figure 2.6 – Exemple d'une approche décentralisée

**Approche hiérarchique** L'approche de supervision hiérarchique de Zhong [Zhong et Wonham 1990] repose sur une conception des modèles à deux niveaux. La Figure 2.7 illustre ce découpage dans lequel le bas niveau est constitué d'un procédé $G_l$ et de son contrôleur $S_l$ et le haut niveau est formé d'un modèle $G_h$ et de son contrôleur $S_h$. $G_l$ est le procédé réel qui sera contrôlé par $S_l$, alors que $G_h$ est un modèle abstrait de $G_l$, utilisé avec son contrôleur $S_h$ pour prendre des décisions plus globales et plus abstraites que les décisions prises au niveau bas. Ces décompositions, horizontales et verticales, permettent de réduire l'explosion combinatoire, en structurant les modèles.

Dans cette section, nous avons présenté les bases de la théorie du contrôle par supervision ainsi les principes de quelques approches cherchant à résoudre les problèmes d'interprétation et de calculabilité liés à la taille des modèles. Cependant, ces approches ne sont pas applicables pour la gestion de modes, car en construisant globalement le procédé G elles considèrent l'utilisation systématique de tous les composants du système.

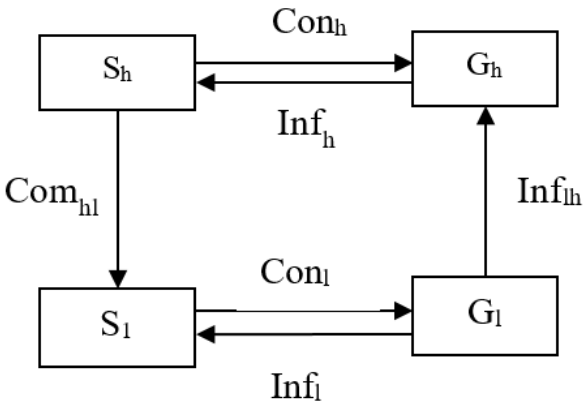



Figure 2.7 – Approche hiérarchique de Zhong

### 2.3.3 Gestion des modes de fonctionnement

En se focalisant sur la maîtrise de comportement des SEDs, une des notions couramment utilisée pour la conception des systèmes est la notion de mode de fonctionnement. Un mode de fonctionnement est un comportement spécifique du système pendant une période de fonctionnement et engageant un ensemble réduit de composants. Il peut correspondre à un changement de configuration (démarrage/arrêt du train, décollage/atterrissage/pilotage automatique d'avion, etc. ) ou à un changement d'objectif (mode nominal, mode dégradé, mode de test). La gestion des modes de fonctionnement a pour rôle de faire correspondre à chaque mode à un comportement spécifique du système et des tâches spécifiques et de contrôler la commutation entre les modes.

#### Gestion des modes de fonctionnement et TCS

De nombreuses recherches ont porté sur la gestion des modes de fonctionnement des SEDs. Certaines de ces recherches se sont reposées sur les approches classiques de conception. Elles ont permis la vérification/validation des comportements internes de chaque mode de fonctionnement séparément «a posteriori» mais la problématique majeure est les commutations entre les modes.
Par exemple, [Adepa 1981, Asarin et al. 2000] se sont concentrés sur les problèmes de caractérisation et de commutation entre les modes. Cependant, leurs approches ne s'appuient sur aucun modèle formel : elles ne possèdent ni mécanisme de validation des alternances possibles (habilitation et validité du changement de mode), ni mécanisme de validation de la recherche par blocage. [Hamani et al.2004] ont proposé une approche dans laquelle la caractérisation et la commutation de mode sont définies mais sans validation formelle du mécanisme de commutation et de l'évitement des blocages.

À l'opposé de ces méthodes, la TCS permet de prendre en compte «a priori» les propriétés attendues du système pour synthétiser l'ensemble des comportements admissibles [Ramadge et Wonham 1987]. Le concepteur garde bien sûr la responsabilité de la spécification de ces propriétés, comme dans les approches classiques, mais une erreur dans ces spécifications peut être détectée plus tôt.

Les travaux de [Nourelfath 1997] ont été les premiers à proposer l'utilisation la TCS pour intégrer, à l'étude du mode nominal, la prise en compte des comportements admissibles du système après une panne d'un de ses composants via un modèle à automate. Cependant, ils ne permettaient d'étudier que deux modes de fonctionnement : le nominal et le mode dégradé. La suppression de cette limitation a été proposée dans les travaux de [Kamach 2004]. Ces travaux ont permis de résoudre le problème lié à l'utilisation de plusieurs modèles. En effet, ils ont proposé une approche dite multi-modèle à base d'automates pour la gestion des modes de fonctionnement. Cette approche permet la représentation d'un système complexe par un ensemble de modèles simples. Chaque modèle est une description partielle du système dans un mode de fonctionnement donné. Dans l'approche de Kamach, un automate est associé à chaque mode de fonctionnement et la commutation d'un mode à un autre est assuré par un mécanisme de suivi qui permet de déterminer l'état de départ du nouveau mode à partir duquel l'événement de commutation s'est produit. Kamach a montré l'unicité du superviseur par mode, où un modèle du procédé et un modèle des spécifications sont associés à chacun des modes de fonctionnement. Le modèle du procédé représente la configuration des éléments assurant la fonctionnalité attendue dans le mode. Le modèle des spécifications exprime les contraintes de fonctionnement imposées au modèle de procédé correspondant.

L'objectif de l'approche multi-modèle est d'enrichir la TCS tout en gardant des modèles de taille raisonnable. Cependant, les procédures de calcul des états de départ et de retour, entre les commutations et entre les modèles des procédés, alourdissent l'approche. Pour résoudre ce problème de taille du modèle, [Kadri et al. 2013] ont appliqué l'approche multi-modèle aux Réseaux de Petri Colorés (RdPCs) [Jensen 1997]. Le principe est resté le même de celui de Kamach : représenter un système complexe par un ensemble de modèles simples. Chaque modèle RdPC représente le comportement exigé du système dans un mode de fonctionnement donné. Le passage d'un mode à l'autre est toujours dû à l'occurrence d'un événement d'exception appelé événement de commutation.

[Kadri et al. 2013] ont mis en place un mécanisme de commutation plus simple entre les différents modes en minimisant la taille du modèle global, et qui facilite aussi l'analyse formelle du système entier. De plus, pour la détermination des états de départ, en passant d'un mode à l'autre, des composants utilisés dans des différents modes de fonctionnement, [Kamach 2004] a mis en place un mécanisme de mémorisation afin de maintenir une trace sur les événements qui se sont produits pour ces composants communs. Ce méca-

nisme de mémorisation est due au fait que l'évolution des états de composants communs n'est pas prise en compte en termes d'automates. Cependant, [Kadri et al. 2013] ont évité cette étape de mémorisation car les composants communs sont modélisés sans duplication dans le modèle RdPC global et par conséquent aucun mécanisme n'est mis en place pour la mémorisation de l'état de ces composants car leur état se conserve implicitement en passant d'un mode à un autre.

De ce fait, et comme le choix du formalisme de modélisation influe considérablement sur l'applicabilité de la gestion de modes de fonctionnement, nous avons choisi le formalisme basé sur les RdPHNs. De plus, lorsque les spécifications sont modélisés, par des automates ou par des RdPs, nous avons deux modèles par mode (un automate modélisant le procédé et un automate ou un RdP modélisant la spécification). Par contre, le choix des RdPHNs comme modèle rend possible la modélisation de chaque mode par un seul modèle RdPHN décrivant le comportement du système et sa couche de contrôle [Kadri 2010] .

### 2.3.4 Réseaux de Petri

Les Réseaux de Petri (RdPs) ont été introduits par Carl Adam Petri en 1962 [Petri 1962]. Ils sont utilisés dans le cadre de la modélisation des SEDs et particulièrement les systèmes de production manufacturiers. Ils permettent de représenter de manière concise dans une structure unifiée à la fois les aspects statiques et dynamiques du système considéré, grâce à leur double représentation, graphique et mathématique. En particulier, l'aspect graphique permet de concevoir et de vérifier le modèle de manière concise, tandis que la description mathématique permet de simuler le système considéré dans des environnements logiciels, en considérant différentes conditions dynamiques.

#### Application des RdPs au système ERTMS

Dans les recherches sur la modélisation des systèmes ferroviaires, les RdPs ont été largement utilisés. [Jabri et al. 2010] ont utilisé les RdPs interprétés et le langage Unified Modelling Language (UML) pour produire des modèles formels de la spécification et générer des scénarios de tests pour la validation de composants ERTMS sur la base d'exigences fonctionnelles. Un scénario est considéré comme une séquence de tir dans le graphique d'atteignabilité du RdP. Ensuite, des scénarios de test sont appliqués sur le simulateur de la plateforme ERTMS afin de vérifier les composants et de donner des verdicts de test. [Amraoui et Mesghouni 2014] ont utilisé les RdPCs pour modéliser l'échange de communication entre un train et son centre de contrôle en mettant en évidence des concepts de la théorie des réseaux comme la concomitance, le conflit et la dépendance causale. [Dhahbi et al. 2012] ont proposé un modèle RdPC du mouvement du train et de

sa localisation sur une voie ferrée équipée du système ERTMS Niveau 2. De plus, leur modèle inclut l'interaction entre le train et les Eurobalises.

### Réseaux de Petri de Haut Niveau

Le problème majeur des RdPs est l'explosion du nombre d'éléments de leur forme graphique lorsqu'ils sont utilisés pour décrire des systèmes complexes. Pour résoudre ce problème, les Réseaux de Petri de Haut Niveau (RdPHNs) [ISO/IEC. 2002] ont été développés en introduisant des concepts de niveau supérieur, tels que l'utilisation de données structurées complexes en tant que jetons et l'utilisation d'expressions algébriques pour annoter les éléments du réseau.
Parmi les formes des RdPHNs, nous pouvons citer les réseaux de Petri prédicats/transitions (RdPP/Ts) [Genrich 1991], les réseaux de Petri colorés (RdPCs), les réseaux de Petri stochastiques (RdPSs) [Florin et Natkin 1985], les réseaux de Petri hiérarchiques (RdPHs) [Jensen et al. 2007] etc.
[Sun et al. 2014] ont introduit un nouveau modèle de réseau de Petri coloré hiérarchique (RdPCH) pour un système de signalisation/enclenchement ferroviaire (RIS). Ils présentent également une étude de cas dans laquelle une zone de gare standard est soumise aux règles de sécurité et enclenchements français en vigueur.
Dans ce qui suit, nous présentons les RdPHNs que nous allons utiliser dans le reste de notre travail.

**Réseaux de Petri colorés** Les Réseaux de Petri Colorés (RdPCs) sont un langage de modélisation et de validation des SEDs dans lequel la concurrence, la communication et la synchronisation jouent un rôle majeur. Ils combinent les réseaux de Petri avec le langage de programmation standard appelé «Méta Language» (ML). Les RdPs constituent le fondement de la notation graphique et des primitives de base pour la modélisation de la simultanéité, de la communication et de la synchronisation. La norme ML fournit les primitives pour la définition des types de données, décrivant la manipulation des données et pour la création de modèles compacts et paramétrables. Ci-dessous, nous rappelons quelques notions de base sur les RdPCs.

**multiset** est un ensemble dans lequel les éléments donnés peuvent apparaître plusieurs fois. Soit un ensemble $A$, $Bag(A)$ désigne l'ensemble des multisets finis sur $A$.

**Classe d'objets** est un ensemble fini et non vide d'objets, également appelé couleurs de base.

**Domaine de couleur** peut être défini comme un produit cartésien de classes d'objets et il est associé à une transition ou à une place. Lorsqu'il est associé à une transition, il

définit l'ensemble de toutes ses instances de tir. Et lorsqu'il est associé à une place, il définit l'ensemble de tous ses marquages possibles.

**Fonctions de couleur** sont une somme pondérée de tuples de fonctions de couleur de base. Les fonctions de couleur sont associées aux étiquettes des arcs de RdPC. Ces fonctions permettent de spécifier les jetons colorés à consommer et à produire lors du franchissement d'une transition donnée.

**garde** est une fonction booléenne définie sur un domaine de couleur dont le rôle est de le limiter à un sous-domaine.

**Définition 2 (Réseau de Petri coloré )**
*Un RdPC est un 8-tuple* $\Sigma =< P, T, K, D, W^-, W^+, \phi, M_0 >$, *où*

$P$ *est un ensemble fini de places ;*

$T$ *est un ensemble de transitions vérifiant* $P \cap T = \emptyset, P \cup T \neq \emptyset$ *;*

$K = \{C_1, ..., C_{|K|}\}$ *est un ensemble de classes d'objets telles que* $\forall i \neq j, C_i \cap C_j = \emptyset$ *;*

$D$ *est la fonction de domaine de couleur, définie par* $P \cup T$ *dans l'ensemble des domaines de couleur. Un élément* $c$ *de* $D(s)$ *est un tuple* $< c_1, ..., c_k >$ *et appelé couleur de* $s$ *;*

$W^-$, $W^+$ *sont les fonctions d'entrée et de sortie (également appelées fonctions d'incidence) définies sur* $P \times T$, *telles que* $W^-(p,t)$ *et* $W^+(p,t)$ *sont des fonctions de couleur représentant des applications linéaires mappant* $Bag(D(t))$ *sur le* $Bag(D(p))$, *pour tout* $(p,t) \in P \times T$. *En d'autres termes,* $W^-(p,t)$ *(respectivement* $W^+(p,t)$*) représente un arc coloré d'entrée (respectivement de sortie) d'un RdPC ;*

$\phi$ *est une fonction qui associe une garde à toute transition. Par défaut,* $\phi(t)$ *est vrai pour toute transition* $t$ *;*

$M_0$ *est le marquage initial. C'est une fonction définie sur* $P$, *telle que* $M_0(p) \in Bag(D(p)), \forall p \in P$.

Le comportement dynamique des RdPCs est déterminé par la règle de tir suivante :
Une transition à guarde $t$ est activée pour une couleur $c$ et un marquage $M$, noté $M[t, c\rangle$, si et seulement si $\forall p \in P, M(p) \geq W^-(p,t)(c)$ et, la garde associée à $t$ est évaluée à vrai.
Le marquage $M'$ obtenu après le tir de $(t, c)$ est calculé comme suit :
$\forall p \in P, M'(p) = M(p) + W^+(p,t)(c) - W^-(p,t)(c)$
La notation $M[t, c > M'$ est utilisée pour indiquer cette relation d'atteignabilité. La notation $[M >$ indique l'ensemble les marquages accessibles à partir du marquage $M$.

**Arcs inhibiteurs** Les arcs inhibiteurs, introduit dans [Agerwala et Flynn], sont une extension qui permet de tester si une place est vide. Un arc inhibiteur est un arc dont extrémité est marquée par un petit cercle qui en absence de jeton dans la place d'entrée sensibilise la transition aval.

**Réseaux de Petri colorés-hiérarchiques** Les Réseaux de Petri Colorés-Hiérarchiques (RdPCHs) permettent au concepteur de construire de grands modèles à partir d'un ensemble de petits modèles RdPCs qui sont liés les uns aux autres de manière bien définie [Jensen et al. 2007, Jensen 2013]. Ceci est similaire à la situation dans laquelle un programmeur construit un grand programme au moyen d'un ensemble de modules.

**Définition 3 (Réseaux de Petri colorés-modulaires)**
*Le RdP colorés-modulaires est un 4-tuple* $\Sigma_M =< \Sigma, T_{sub}, P_{port}, PT >$, *où*

$\Sigma =< P, T, K, D, W^-, W^+, \phi, M_0 >$ ;

$T_{sub}$ *est un ensemble de transitions de substitution ;*

$P_{port}$ *est un ensemble de places de fusions ;*

$PT : P_{port} \rightarrow \{IN, OUT, I/O\}$.

**Définition 4 (Réseaux de Petri colorés hiérarchiques)**
*Un RdPCH est un 4-tuple* $\Sigma_H =< S, SM, PS, FS >$, *où*

$S$ *est un ensemble fini de modules. Chaque module est un réseau de Petri coloré-modulaire* $s =<< P_s, T_s, K_s, D_s, W_s^-, W_s^+, \phi_s, M_{s,0} >, T_{sub,s}, P_{port,s}, PT_s >$ *tel que* $(P_{s_1} \cup T_{s_1}) \cap (P_{s_2} \cup T_{s_2}) = \emptyset$ *pour tout* $s_1, s_2 \in S \setminus s_1 \neq s_2$ ;

$SM : T_{sub} \rightarrow S$ *est une fonction de sous-module qui assigne un sous-module à chaque transition de substitution. Il est requis que la hiérarchie des modules soit acyclique ;*

$PS$ *est une fonction de relation port-socket qui assigne une relation port-socket* $PS(t) \subseteq P_{sock}(t) \times P_{port}^{SM(t)}$ *à chacun transition de substitution* $t$ *tels que* $PT(P) = PT(P'), D(p) = D(p')$ *et* $M_0(p) = M_0(p'0)$ *pour tous* $(p, p') \in PS(t)$ *et tous* $t \in T_{sub}$ ;

$FS \subseteq 2^P$ *est un ensemble des ensembles de fusion non vides tels que* $D(p) = D(p')$ *et* $M_0(p) = M_0(p'0)$ *pour tous* $p, p' \in fs$ *et tous* $fs \in FS$.

**Réseaux de Petri colorés à priorité** Les Réseaux de Petri Colorés à Priorité (RdPCP) sont une extension des réseaux de Petri colorés [Jensen 1997, Jensen et al. 2007] où les transitions peuvent avoir des priorités. Les priorités peuvent être utilisées pour établir un ordre de tir. Plus précisément, nous utilisons les priorités suivantes : `P_HIGH, P_NORMAL` et `P_LOW`, en respectant cet ordre de priorité décroissant.

**Définition 5 (Réseaux de Petri colorés à priorité)**
*Un RdPCP est un 2-tuple* $\Sigma_P =< \Sigma, \Pi >$, *où*

$\Sigma = < P_s, T_s, K_s, D_s, W_s^-, W_s^+, \phi_s, M_{s,0} >$

$\Pi : T \rightarrow \mathbb{N}$ *est la fonction de priorité. Elle attribue un niveau de priorité à chaque transition, où les valeurs faibles correspondent aux priorités élevées. Nous prenons* 1

*comme niveau de priorité d'une transition avec la priorité* `P_HIGH`*,* 2 *pour* `P_NORMAL` *et* 3 *pour* `P_LOW`*.*

Tous les modèles RdPHNs utilisés dans ce travail sont modélisés en utilisant l'environnement [CPN Tools].

**CPN Tools**

CPN Tools est un outil informatique puissant pour la construction et l'analyse de modèles RdPs développés par le groupe CPN de l'Université d'Aarhus, Danemark. À l'aide de CPN Tools, il est possible d'étudier le comportement du système modélisé à l'aide de la simulation, de vérifier les propriétés à l'aide de méthodes d'espace d'états et de vérification du modèle, et d'effectuer une analyse de performance basée sur la simulation. L'interaction de l'utilisateur avec CPN Tools est basée sur la manipulation directe de la représentation graphique du modèle RdPC à l'aide de techniques d'interaction, telles que les palettes d'outils et les menus de marquage.
Le simulateur de CPN Tools permet d'exploiter un nombre important des structures de données et d'algorithmes pour une simulation efficace de grands modèles RdPCHs [Mortensen 2001].

## 2.4 Conclusion

Ce chapitre a introduit les bases de notre travail. Dans un premier temps, nous avons présenté le système ERTMS, qui est notre application cible, ainsi que ses composants, ses niveaux de fonctionnement et ses modes opératoires. Dans un deuxième temps, nous avons énoncé la problématique scientifique liée au franchissement de frontière en utilisant ERTMS. Dans un troisième temps, nous avons passé en revue les différentes approches qui nous ont permis de concevoir des modèles sûrs par construction pour les SdSs à savoir l'approche multi-modèle et la TCS. Dans un dernier temps, nous avons présenté le formalisme de modélisation qui est les RdPHNs et l'outil de modélisation CPN Tool.

# Chapitre 3

# Démarche de conception des systèmes complexes

## 3.1 Introduction

Ce chapitre présente une démarche hiérarchique pour le contrôle et le suivi de la conception des systèmes complexes dynamiques dont la structure change au cours du temps. Le fonctionnement de ces systèmes est décrit par l'approche multi-modèle qui utilise le concept de mode de fonctionnement qui représente un fonctionnement non permanent du système dans une configuration physique particulière. En adoptant une approche «bottom-up» (en littérature de bas en haut), la démarche proposée répond à la problématique de gestion de mode et porte principalement sur la conception de modèles sûrs par construction et sur leurs commutations tout en tenant compte des actions de reconfigurations, de la gestion des états de départ et des ressources communes.

## 3.2 Exemple directeur

### 3.2.1 Description du système

Le système étudié, illustré par la Figure 3.1, est un système de production. Il comprend quatre machines de production $C1$, $C2$, $C3$, $C4$ et un stock intermédiaire $B$ de capacité 3. Les stocks amont et aval à ce système ne sont pas considérés. Le système comporte deux modes de fonctionnement, un mode nominal et un mode dégradé. Dans ce système, toutes les machines fonctionnent de façon indépendante, puisent des pièces brutes en amont et rejettent des pièces usinées en aval. La machine $C2$ est redondante avec la machine $C1$. Les machines $C1$ et $C2$ reçoivent des pièces brutes d'un stock en amont de capacité supposée infinie. Après traitement, la pièce est ensuite déposée dans le stock $B$ qui est incrémenté

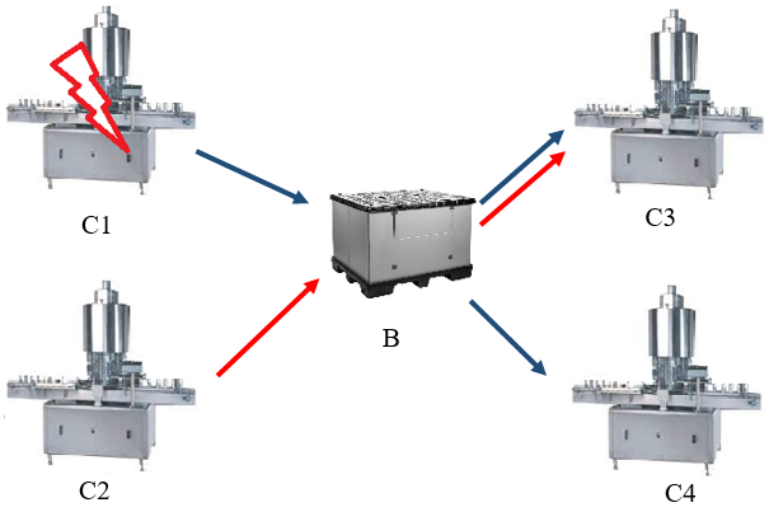



Figure 3.1 – Système de production

d'une unité. Lorsqu'une pièce est présente dans le stock $B$, une tâche peut être lancée sur la machine $C3$ ou sur la machine $C4$. Le lancement d'une tâche sur l'une de ces deux machines décrémente le stock $B$ d'une unité.

### 3.2.2 Décomposition modale

Nous avons décrit le système physique et les relations entre composants. Maintenant nous allons décrire les différents modes de fonctionnement du système. Le système possède un mode nominal (*Nominal*) dans lequel la machine $C2$ ne doit pas produire. La machine $C1$ produit des pièces une par une placées ensuite dans le stock $B$. Ces pièces sont ensuite consommées par la machine $C3$ ou la machine $C4$, sans notion de priorité, qui produisent chacune des pièces.
Le système peut tolérer une panne au niveau de la machine $C1$ lorsqu'elle est en marche. Cette panne est symbolisée par l'événement *Failure*. L'occurrence de cet événement implique une commutation du système du mode nominal vers un mode dégradé (*Degraded*) dans lequel la machine $C1$, en panne, n'est plus utilisée jusqu'à l'occurrence de l'événement *Recovery* qui implique le retour du système dans le mode nominal.
Dans le mode *Degraded*, la machine $C1$ est remplacée par la machine $C2$. Cependant, la machine $C2$ est moins rapide que la machine $C1$, donc il n'est plus nécessaire d'utiliser deux machines ($C3$ et $C4$) pour consommer les pièces du stock $B$. De ce fait, seule la machine $C3$ reste active pour produire et la machine $C4$ passe au repos tant que le système est dans le mode dégradé.

## 3.3 Présentation de la démarche

La démarche proposée repose sur l'utilisation de la TCS et offre un contrôle hiérarchique de conception (voir Figure 3.2). Cette démarche comporte trois étapes de construction des modèles.

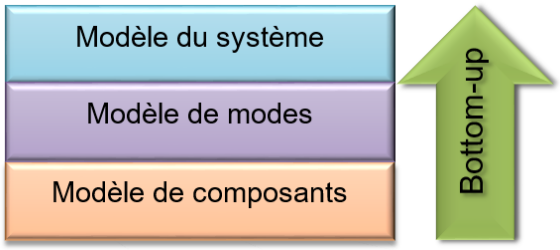



Figure 3.2 – Modèle hiérarchique de conception

« Modèle de composants » est la première étape. Elle permet de concevoir les composants dans des feuilles RdPC séparées. Chaque modèle est vérifié par rapport à la spécification séparément des autres.
La deuxième étape est « Modèle de modes ». Elle consiste à étudier indépendamment et séparément chaque mode de fonctionnement en appliquant de manière conventionnelle la TCS. Chaque mode est caractérisé par un ensemble d'exigences modélisées indépendamment des exigences des autres modes et qui doivent être remplies lorsque le système fonctionne dans ce mode.
La modélisation des modes de fonctionnement se base sur l'approche multi-modèle, dans laquelle un modèle RdPC différent est associé à chaque mode de fonctionnement. Ceci permet de définir un comportement distinct pour chaque modèle et une stratégie de contrôle différente pour chaque mode. Les modèles RdPCs obtenus résultent d'une composition des modèles RdPCs des composants utilisés dans ce mode.
La troisième étape, « Modèle du système », permet de modéliser l'ensemble du système avec les actions de reconfiguration. Tout d'abord, chaque mode comporte un ensemble de composants générant des événements impliquant une commutation. Ainsi, les modèles de mode sont étendus en prenant en compte les transitions de commutation : cela permet de prendre en compte la spécification intermodale. Ensuite, un algorithme de fusion est appliqué pour fusionner les modèles, en évitant la duplication de composants communs. Enfin, un mécanisme de commutation est intégré permettant la commutation entre les différents modes de fonctionnement.

## 3.4 Modèle de composants

Un système est constitué d'un ensemble de composants physiques. Ces composants sont reliés entre eux afin de créer des fonctions spécifiques. Ces composants peuvent tomber en panne et être activés/désactivés selon la tâche à réaliser. Ceci amène à caractériser un ensemble de composants utilisé par le système. Chaque composant peut inclure des évènements de reconfiguration telles que activations, désactivations, panne et reprises et il contient des ports de communication lui permettant de se relier aux autres composants du système. Son fonctionnement est toujours le même indépendamment des modes du système.

**Définition 6 (Ensemble de composants)**
*L'ensemble des composants est appelé* $C = \{C_1, C_2, .., C_{|C|}\}$ *où* $|C| > 1$.

Le système étudié possède quatre machines et un stock. Chaque machine est considérée ici comme un composant. Soit $C = \{C_1, C_2, C_3, C_4\}$ l'ensemble des composantes correspondant aux quatre machines. Le stock n'est pas considéré comme un composant mais plutôt comme une spécification du système. Cela correspond à un choix classique de contrainte de la TCS [Ramadge et Wonham 1989] car le stock n'est pas générateur d'événements qui lui sont propre.

**Définition 7 (Modèle d'un composant)**
*Soit* $C$ *l'ensemble des composants.*
*Un composant* $C_i, i \in \{1..|C|\}$ *est modélisé par un RdPC*
$< P_{C_i}, T_{C_i}, K_{C_i}, D_{C_i}, W^-_{C_i}, W^+_{C_i}, \phi_{C_i}, M_{0,C_i} >$ *où :*

$P_{C_i} = P^{\circlearrowleft}_{C_i} \cup P^{\leftrightarrows}_{C_i}$ *avec* $P^{\circlearrowleft}_{C_i} \cap P^{\leftrightarrows}_{C_i} = \emptyset$. $P^{\circlearrowleft}_{C_i}$ *et* $P^{\leftrightarrows}_{C_i}$ *sont, respectivement, les places internes et les places de communication (partagées) du composant* $C_i$.

$T_{C_i} = T^{\circlearrowleft}_{C_i} \cup T^{\leftrightarrows}_{C_i}$ *avec* $T^{\circlearrowleft}_{C_i} \cap T^{\leftrightarrows}_{C_i} = \emptyset$. $T^{\circlearrowleft}_{C_i}$ *et* $T^{\leftrightarrows}_{C_i}$ *sont, respectivement, les transitions internes et les transitions de commutation du composant* $C_i$.

$W^-_{C_i} = W^{-\circlearrowleft}_{C_i} \cup W^{-\leftrightarrows}_{C_i}$ *avec* $W^{-\circlearrowleft}_{C_i} \cap W^{-\leftrightarrows}_{C_i} = \emptyset$. $W^{-\circlearrowleft}_{C_i}$ *et* $W^{-\leftrightarrows}_{C_i}$ *sont, respectivement, les arcs internes et les arcs de commutation (liés aux transitions de commutation) du composant* $C_i$.

$W^+_{C_i} = W^{+\circlearrowleft}_{C_i} \cup W^{+\leftrightarrows}_{C_i}$ *avec* $W^{+\circlearrowleft}_{C_i} \cap W^{+\leftrightarrows}_{C_i} = \emptyset$. $W^{+\circlearrowleft}_{C_i}$ *et* $W^{+\leftrightarrows}_{C_i}$ *sont, respectivement, les arcs internes et les arcs de commutation du composant* $C_i$.

Dans le cas de notre exemple, le fonctionnement de $C_1$ est représenté par la machine à états (Figure 3.3) composée des places $P^{\circlearrowleft}_{C_i} = \{Init\ C1, Traitment\ C1\}$, transitions $T^{\circlearrowleft}_{C_i} = \{Start\ C1,\ End\ C1\}$, et les arcs associés étiquetés par la variable $P1$ qui est définie sur la classe de couleur $M1 = C1$. $M_0(Init\ C1) = 1`P1$ (un jeton $P1$). Les places

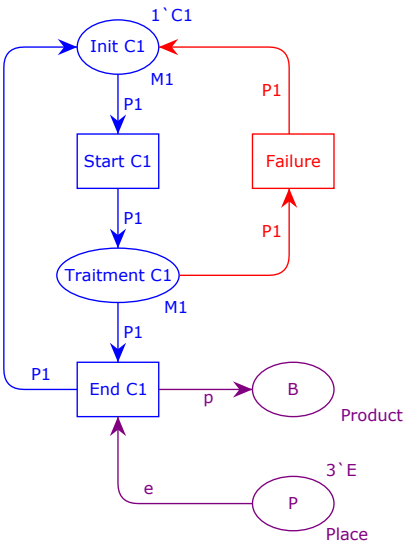



Figure 3.3 – Modèle de $C_1$

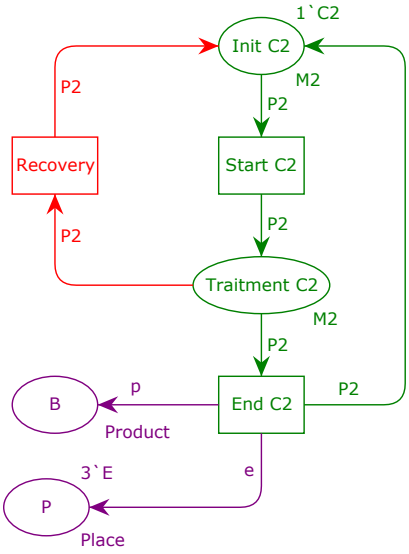



Figure 3.4 – Modèle de $C_2$

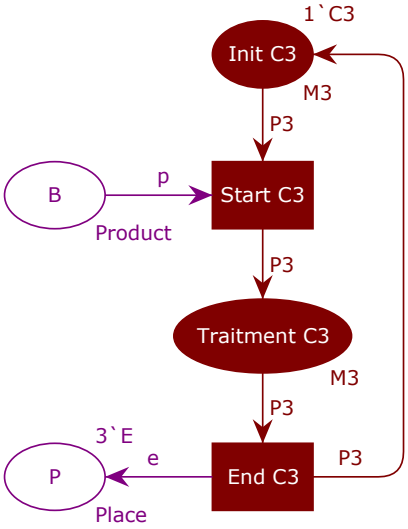



Figure 3.5 – Modèle de $C_3$

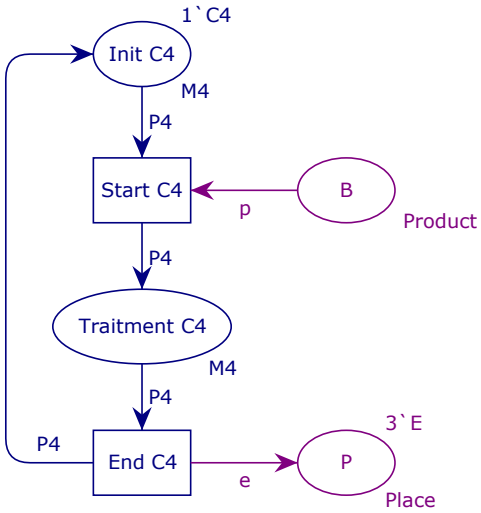



Figure 3.6 – Modèle de $C_4$

$P_{C_1}^{\leftrightarrows} = \{P, B\}$ et les arcs associés représentent le stock intermédiaire lié $B$ de capacité limité à 3. Le stock $B$ a pour but de stocker les pièces produites par $C_1$ et $P$ représente la capacité disponible du stock. Lorsqu'il est déclenché, $C_1$ est défini sur son état $Init\ C1$. $T_{C_1}^{\leftrightarrows} = \{Failure\}$, est une transition de commutation. Lorsqu'elle est tirée, $C_1$ passe à son état $Init\ C1$.

Le fonctionnement de $C_2$ (Figure 3.4) est représenté par les places $P_{C_i}^{\circlearrowleft} = \{Init\ C2, Traitment\ C2\}$, les transitions $T_{C_i}^{\circlearrowleft} = \{Start\ C2, End\ C2\}$ et les arcs étiquetés par $P2$ ($P2$ définit sur $M2 = \{C2\}$). La transition $Recovery$ ($T_{C_2}^{\leftrightarrows} = \{Recovery\}$) est une transition qui permet au système de revenir à son mode nominal. Les places $P_{C_2}^{\leftrightarrows} = \{P, B\} = P_{C_1}^{\leftrightarrows}$ représentent l'état du stock et sont des places de communication.

Le comportement de $C_3$ (Figure 3.5) (resp. $C_4$ (Figure 3.6)) est représenté par les places $P^{\circlearrowleft}_{C_3} = \{Init\ C3, Traitment\ C3\}$ (resp. $P^{\circlearrowleft}_{C_4} = \{Init\ C4, Traitment\ C4\}$), les transitions $T^{\circlearrowleft}_{C_3} = \{Start\ C3, End\ C3\}$ (resp. $T^{\circlearrowleft}_{C_4} = \{start\ C4, End\ C4\}$ ) et les arcs étiquetés par $P3$ ($P3$ définit sur $M3 = P3$) (resp. $P4$ ($P4$ définit sur $M4 = P4$)). $M_0(Init\ C3) = 1`C3$ (resp. $M_0(Init\ C4) = 1`C4$). Les places $P^{\leftrightarrows}_{C_3} = \{P, B\} = P^{\leftrightarrows}_{C_1}$ (resp. $P^{\leftrightarrows}_{C_4} = \{P, B\} = P^{\leftrightarrows}_{C_1}$) représentent aussi l'état du stock et ils sont les places de communication.

## 3.5 Modèle de modes

L'objectif de cette deuxième étape est de permettre une première conception de chaque mode, indépendamment des autres. Cette conception des modes est basé sur une étude intramodale, similaire à l'approche décentralisée présentée dans la TCS, afin de de vérifier que les spécifications à respecter sont bien construites et de n'autoriser que le comportement interne désiré. L'étude se fait sur une configuration particulière du système qui exploite un ensemble de composants : le mode de fonctionnement. Ceci permet une interprétation plus aisée du modèle par le concepteur et facilitant ainsi les corrections éventuelles.

**Définition 8 (Ensemble de modes)**
*L'ensemble des modes de fonctionnement est appelé* $OM = \{OM_1, OM_2, ..., OM_{|OM|}\}$, *où* $|OM| > 1$.
$OM_1$ *est supposé être, par convention, le mode initialement activé.*

Dans notre exemple, le système possède, comme décrit dans le cahier des charges, deux modes, un nominal et un dégradé. Nous définissons alors l'ensemble des modes $OM = \{Nominal, Degraded\}$.

**Définition 9 (Modèle de mode)**
*Soit* $OM$ *l'ensemble des modes de fonctionnement.*
*Un mode* $OM_i, i \in \{1..|OM|\}$ *est défini par un modèle RdPC*
$< P_{OM_i}, T_{OM_i}, K_{OM_i}, D_{OM_i}, W^-_{OM_i}, W^+_{OM_i}, \phi_{OM_i}, M_{0,OM_i} >$.

Pour des raisons de simplicité, nous confondons la notation $OM_i$ d'une identité de mode de fonctionnement et son modèle RdPC associé.

À partir de la spécification, chaque mode définit l'ensemble des composants nécessaires à l'exécution de ses tâches. Et comme tous les composants ne sont pas utilisés dans chaque mode de fonctionnement, alors si un composant est utilisé dans plusieurs modes de fonctionnement, il est appelé un *composant commun*, sinon il s'agit d'un *composant propre*.
La Définition 10 présente les conditions à accomplir par n'importe quel composant $C_i$ pour être commun.

**Définition 10 (Composant commun)**
*Soit $OM$ l'ensemble des modes de fonctionnement.*
$\forall(OM_i, OM_j) \in OM \times OM(OM_i \neq OM_j)$,
*Si* $(P_{C_i} = (P_{OM_i} \cap P_{OM_j}) \neq \emptyset)$ *ou* $(T_{C_i} = (T_{OM_i} \cap T_{OM_j}) \neq \emptyset)$ *alors*
$C_i$ *est un composant commun aux deux modes* $OM_i$ *et* $OM_j$ *tel que*

1. $\forall(p,t) \in P_{C_i} \times T_{C_i}, W^-_{OM_i} = W^-_{OM_j}$ *et* $W^+_{OM_i} = W^+_{OM_j}$ ;
2. $\forall t \in T_{C_i}, \phi_{OM_i}(t) = \phi_{OM_j}(t)$ ;
3. $\forall p \in P_{C_i}, M_{0,OM_i} = M_{0,OM_j}$.

L'étape Modèle de modes consiste à représenter le comportement interne de chaque mode de fonctionnement suite à une décomposition modale des composants utilisés dans ce mode (voir exemple de Figure 3.7). La communication entre composants est assurée par la fusion des places de communication ayant les mêmes noms dans différents modèles de composants.

**Définition 11 (Décomposition de mode (1) )**
*Soit $OM_i$ un mode de fonctionnement.*
*L'ensemble des composants utilisés dans le mode $OM_i$ est appelé* $C_{OM_i} = C^{\circlearrowleft}_{OM_i} \cup C^{\Leftrightarrow}_{OM_i} \cup C^{\leftrightarrows}_{OM_i}$

- $C^{\circlearrowleft}_{OM_i}$ *est l'ensemble des composants propres de* $C_{OM_i}$ ;
- $C^{\Leftrightarrow}_{OM_i}$ *est l'ensemble des composants communs à* $C_{OM_i}$ *et à d'autres modes ;*
- $C^{\leftarrow}_{OM_i}$ *est l'ensemble des composants qui conduisent le système à entrer en mode* $OM_i$ ;
- $C^{\rightarrow}_{OM_i}$ *est l'ensemble des composants qui conduisent le système à quitter le mode* $OM_i$ ;
- $C^{\leftrightarrows}_{OM_i} = C^{\rightarrow}_{OM_i} \cup C^{\leftarrow}_{OM_i}$ *est l'ensemble des composants de commutation.*

**Définition 12 (Décomposition de mode (2))**
*Soit $OM_i$ un mode de fonctionnement.*
*L'ensemble des composants utilisés privés de leurs transitions de commutation dans le mode $OM_i$ est appelé* $\overline{C_{OM_i}} = C^{\circlearrowleft}_{OM_i} \cup C^{\Leftrightarrow}_{OM_i} \cup \overline{C^{\leftrightarrows}_{OM_i}}$ *tels que :*

- $\overline{C^{\leftarrow}_{OM_i}}$ *est l'ensemble des composants qui conduisent le système à entrer en mode* $OM_i$ *privés de leurs transitions de commutation ;*
- $\overline{C^{\rightarrow}_{OM_i}}$ *est l'ensemble des composants qui conduisent le système à quitter le mode* $OM_i$ *privés de leurs transitions de commutation ;*
- $\overline{C^{\leftrightarrows}_{OM_i}} = \overline{C^{\leftarrow}_{OM_i}} \cup \overline{C^{\rightarrow}_{OM_i}}$.

**Remarque 1** *L'union de l'ensemble de tous les composants contenus dans les modes est égale à l'ensemble des composants :* $C = \bigcup_{C_{OM_i}, i \in |OM|}$.

Appliquées à notre exemple (Figure 3.1), ces définitions nous donnent la Figure 3.7 qui illustre une décomposition modale. Sur cette figure, chaque composant est représenté dans le mode dans lequel il est utilisé.

$OM = \{Nominal, Degraded\}$
$\overline{C_{Nominal}} = \{C_1, C_2, C_3\}$
$\overline{C_{Degraded}} = \{C_3, C_2\}$
$C^{\circlearrowleft}_{Nominal} = \{C_1, C_4\}$
$C^{\circlearrowleft}_{Degraded} = \{C_2\}$
$\overline{C^{\Leftrightarrow}_{Nominal,Degraded}} = \{C_3\}$
$\overline{C^{\leftrightarrows}_{Nominal}} = \{C_1\}$
$\overline{C^{\leftrightarrows}_{Degraded}} = \{C_2\}$

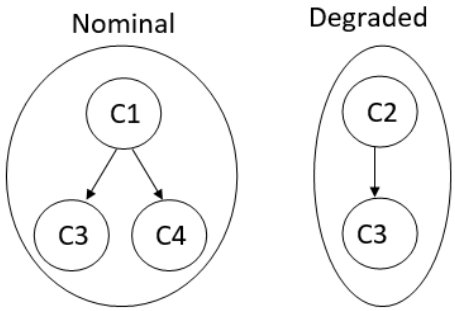



Figure 3.7 – Exemple d'une décomposition modale

Nous pouvons voir sur cette figure que le mode *Nominal* est composé des trois composants $C_1$, $C_3$ et $C_4$ et que le mode *Degraded* possède deux composants $C_2$ et $C_3$. Nous pouvons aussi remarquer qu'il existe un composant commun aux modes *Nominal* et *Degraded* : $C_3$. Les composants communs sont dupliqués à cette étape dans chaque modèle correspondant mais cette duplication sera traitée dans l'étape modèle du système.

Pour obtenir des modèles de modes fiables et bien construits conformément aux exigences, nous présentons un algorithme (voir Algorithme 1) générant les modèles RdPCs des modes de fonctionnement à partir des modèles de composants qui les composent.

En appliquant l'Algorithme 1 à notre exemple, nous obtenons le modèle RdPC du mode *Nominal* illustré par la Figure 3.8 et du mode *Degraded* illustré par la Figure 3.9. Néanmoins, ces modèles ne sont pas interconnectés, et c'est l'objectif principal de la section suivante.

## 3.6 Modèle du système

L'objectif de cette section est d'unir les modèles de modes afin d'obtenir un modèle unique où seules les commutations désirées entre modes peuvent se produire. Pour cela, nous allons d'abord étendre les modes $OM_i$ en ajoutant des transitions de commutation à l'ensemble composants de commutation. Ensuite, une fonction de fusion est proposée permettant la fusion des modèles de modes en un seul modèle tout en supprimant les redondances issue des modèles de composants communs. Enfin, un mécanisme de commutation est intégré permettant de basculer entre les modes de fonctionnement lorsqu'un événement de commutation survient (i.e.défaillance, récupération de composant, etc.).

**Algorithme 1 :** Génération des modèles de modes

**Entrées :** l'ensemble $OM$ de modes de fonctionnement ;
l'ensemble des composants $C_j, (j = 1...|C|)$ ;
**Output :** l'ensemble des RdPCs $OM_i, (i = 1...|OM|)$

**pour chaque** *mode de fonctionnement* $OM_i, i \leftarrow 1$**à** $|OM|$ **faire**
  $P_{OM_i} \leftarrow \varnothing$;
  $T_{OM_i} \leftarrow \varnothing$;
  $K_{OM_i} \leftarrow \varnothing$;
  **pour chaque** *composant* $C_j, i \in \overline{C_{OM_i}}$ **faire**
    $P_{OM_i} \leftarrow P_{OM_i} \cup P^{C_j}$;
    $T_{OM_i} \leftarrow T_{OM_i} \cup T_{C_j}$;
    $K_{OM_i} \leftarrow K_{OM_i} \cup K_{C_j}$;
    **pour chaque** $(p,t) \in P^{C_j} \times T^{C_j}$ **faire**
      $W^-_{OM_i}(p,t) \leftarrow W^-_{C_j}(p,t)$;
      $W^+_{OM_i}(p,t) \leftarrow W^+_{C_j}(p,t)$ ;
    **fin**
    $\phi_{OM_i}(t) \leftarrow \phi_{C_j}(t)$;
  **fin**
  **pour chaque** $p \in P_{C_j}$ **faire**
    $M_{0,OM_i}(p) \leftarrow M_{0,C_j}(p)$ ;
  **fin**
  **pour chaque** *component* $C_k, k \leftarrow 1$**à** $(j-1)$ **faire**
    **pour chaque** $(p,t) \in P_{C_j} \times T_{C_k}, p \notin P_{C_k}, t \notin T_{C_j}$ **faire**
      $W^-_{OM_i}(p,t) \leftarrow 0$;
      $W^+_{OM_i}(p,t) \leftarrow 0$ ;
    **fin**
  **fin**
**fin**

### 3.6.1 Extension des modèles de modes

Dans l'étape Modèle de modes, les modes de fonctionnement $OM_i$ sont construits à partir d'une décomposition des composants sans les transitions de commutation pour représenter le comportement interne du système dans ces modes. Ceci permet de s'assurer que le comportement interne des modes respecte les spécifications du système.
Pour représenter le comportement interne et externe des modes, une extension des modèles de modes de fonctionnement est alors nécessaires pour prendre en compte les commutations possibles entre les modes. Cette extension nécessite l'ajout des transitions de commutation à l'ensemble des composants de commutation. L'Algorithme 2 permet d'étendre les modèles RdPCs par ajout des transitions de commutations et aussi leurs arcs correspondants.

En appliquant l'Alorigthme 2 à notre exemple, nous obtenons alors les modèles des Figures 3.10 et 3.11. Nous remarquons que les transitions de commutation *Failure* et

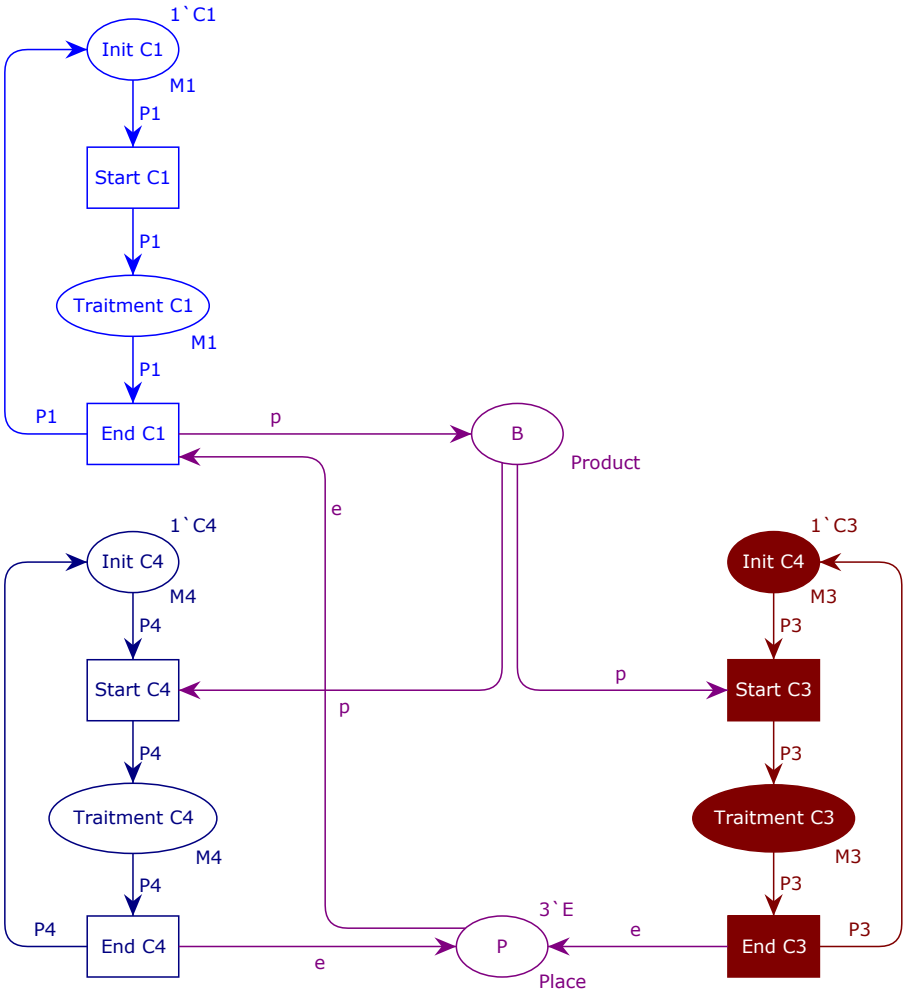



Figure 3.8 – Modèle du mode *Nominal*

*Recovery* sont insérées dans leurs modèles de modes correspondant. Ces transitions seront utiles pour le mécanisme de commutations afin de modéliser la dynamique du système.

### 3.6.2 Fusion des modèles de modes étendus

La fusion des modes, définit par l'Algorithme 3, permet d'obtenir un modèle unique à partir des différents modèles de mode de fonctionnement. Pour assurer un état cohérent des composants communs lors de la commutation de modes, les modèles des composants communs sont fusionnés. L'évolution du marquage des composants communs est implicitement gérée par le graphe de marquage de l'accessibilité des RdPs. En plus, cette fusion permet de réduire la complexité et la taille du modèle final.

Après l'application de l'Algorithme 3 aux modèles de modes de notre exemple, nous obtenons le modèle illustré par la Figure 3.12.

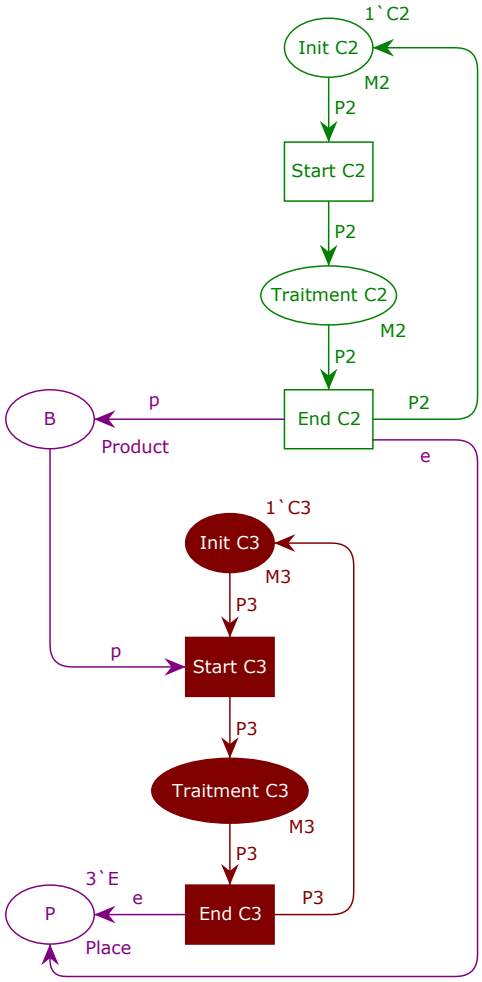



Figure 3.9 – Modèle du mode *Degraded*

### 3.6.3 Intégration du mécanisme de commutation

Cette sous-section se concentre sur le comportement intermodal et prend en compte le comportement pouvant conduire à une commutation entre les modes.
Premièrement, une décomposition multi-modèle du système est réalisée dans laquelle les relations entre les modes sont représentées. La Figure 3.13 illustre la décomposition multi-modèle du système étudié. Nous pouvons voir sur cette figure qu'à partir du mode *Nominal*, un basculement vers le mode *Degraded* est possible grâce à l'événement commutateur *Failure* généré par le composant $C_1$. Dans ce cas, le composant $C_2$ est activé pour remplacer $C_1$ et le composant $C_4$ passe à l'état de repos alors que le composant $C_3$ continue à fonctionner puisque c'est un composant commun aux deux modes. Quand $C_2$ génère l'événement *Recovery*, le système retourne au mode *Nominal*.

Deuxièmement et comme indiqué précédemment, le mécanisme de commutation se déclenche lorsqu'un événement de commutation se produit, tel qu'une défaillance ou une récupération de composant. Il induit la désactivation du mode de fonctionnement actuel

**Algorithme 2 :** Extention des modèles de modes

**Entrées :** l'ensemble $OM$ de modes de fonctionnement ;
l'ensemble des RdPCs $OM_i, (i = 1...|OM|)$ ;
**Output :** l'ensemble des RdPCs $OM_i, (i = 1...|OM|)$

**pour chaque** *mode de fonctionnement* $OM_i, i \leftarrow 1$**à** $|OM|$ **faire**
  **pour chaque** *composant* $C_j, i \in C_{OM_i}$ **faire**
    $T_{OM_i} \leftarrow T_{OM_i} \cup T_{C_j}$;
    **pour chaque** $(p,t) \in P_{C_j} \times T_{C_j}$ **faire**
      $W^-_{OM_i}(p,t) \leftarrow W^-_{C_j}(p,t)$;
      $W^+_{OM_i}(p,t) \leftarrow W^+_{C_j}(p,t)$ ;
    **fin**
    $\phi_{OM_i}(t) \leftarrow \phi_{C_j}(t)$;
  **fin**
**fin**

et l'activation d'un mode de destination. Dans notre approche, le mécanisme de commutation est modélisé par les transitions $T^{\hookleftarrow}_{C_i}$ dans le mode correspondant. Le déclenchement d'une telle transition doit désactiver les transitions du mode source et permettre le déclenchement des transitions du mode destination. Pour distinguer ce type de transitions, nous définissons une application notée *target_mode* dont le rôle est d'associer à chaque transition son mode de destination.

**Définition 13 (Mécanisme de commutation)**
*Soit $OM_i$ un mode de fonctionnement et $T_{OM_i}$ son ensemble de transitions associé.*
*Soit $target\_mode : T_{OM_i} \rightarrow OM$ une application tel que $target\_mode(t)$ indique le mode de fonctionnement actif après le déclenchement de $t$.*

**Remarque 2** *$\forall t \in T_{OM_i}$, si $target\_mode(t) \neq OM_i$, alors $t$ correspond à un mécanisme de commutation du mode $OM_i$ menant au mode $target\_mode(t)$.*

Pour assurer cette fonction, nous proposons d'identifier chaque modèle $OM_i$ par une couleur bien déterminée qu'on le note par $om_i$.

**Définition 14 (Couleur de modèle)**
*On appelle Col la bijection qui associe à chaque modèle RdPC $OM_i$ une couleur $om_i$ tel que*

$$Col : \bigcup OM_i \rightarrow mode$$

$$OM_i \longmapsto om_i$$

*où mode un ensemble fini et non vides de couleurs.*

**Algorithme 3 :** Fusion des modes

**Entrées :** l'ensemble $OM$ de modes de fonctionnement ;
l'ensemble des RdPCs $OM_i, (i = 1...|OM|)$ ;
**Output :** un modèle RdPC $G =< P, T, K, D, W^-, W^+, \phi, M_0 >$

$P \leftarrow \varnothing$;
$T \leftarrow \varnothing$;
$K \leftarrow \varnothing$;
**pour chaque** *mode de fonctionnement* $OM_i, i \leftarrow 1$**à** $|OM|$ **faire**
  $P \leftarrow P \cup P_{OM_i}$;
  $T \leftarrow T \cup T_{OM_i}$;
  $K \leftarrow K \cup K_{OM_i}$;
  **pour chaque** $(p,t) \in (P_{OM_i} \setminus P_{C^{\Leftrightarrow}_{OM_i}}) \times (T_{OM_i} \setminus T_{C^{\Leftrightarrow}_{OM_i}})$ **faire**
    $W^-(p,t) \leftarrow W^-_{OM_i}(p,t)$;
    $W^+(p,t) \leftarrow W^+_{OM_i}(p,t)$;
  **fin**
  **pour chaque** $(p,t) \in P_{C^{\Leftrightarrow}_{OM_i}} \times T_{C^{\Leftrightarrow}_{OM_i}}$ **faire**
    **si** $W^-(p,t) = 0$ **alors**
      $W^-(p,t) \leftarrow W^-_{OM_i}(p,t)$;
    **fin**
    **si** $W^+(p,t) = 0$ **alors**
      $W^+(p,t) \leftarrow W^+_{OM_i}(p,t)$;
    **fin**
  **fin**
  **pour chaque** $t \in T_{OM_i}$ **faire**
    $\phi(t) \leftarrow \phi_i(t)$ ;
  **fin**
  **pour chaque** $p \in P_{OM_i}$ **faire**
    $M_0(p) \leftarrow M_{0,OM_i}(p)$ ;
  **fin**
  **pour chaque** $p \in P^{C_j}$ **faire**
    $M_{0,OM_i}(p) \leftarrow M_{0,C_j}(p)$ ;
  **fin**
  **pour chaque** *mode* $OM_j, i \leftarrow 1$**à** $(i-1)$ **faire**
    **pour chaque** $(p,t) \in P_{OM_i} \times T_{OM_j}, p \notin P_{OM_j}, t \notin T_{OM_i}$ **faire**
      $W^-_{OM_i}(p,t) \leftarrow 0$;
      $W^+_{OM_i}(p,t) \leftarrow 0$ ;
    **fin**
  **fin**
**fin**

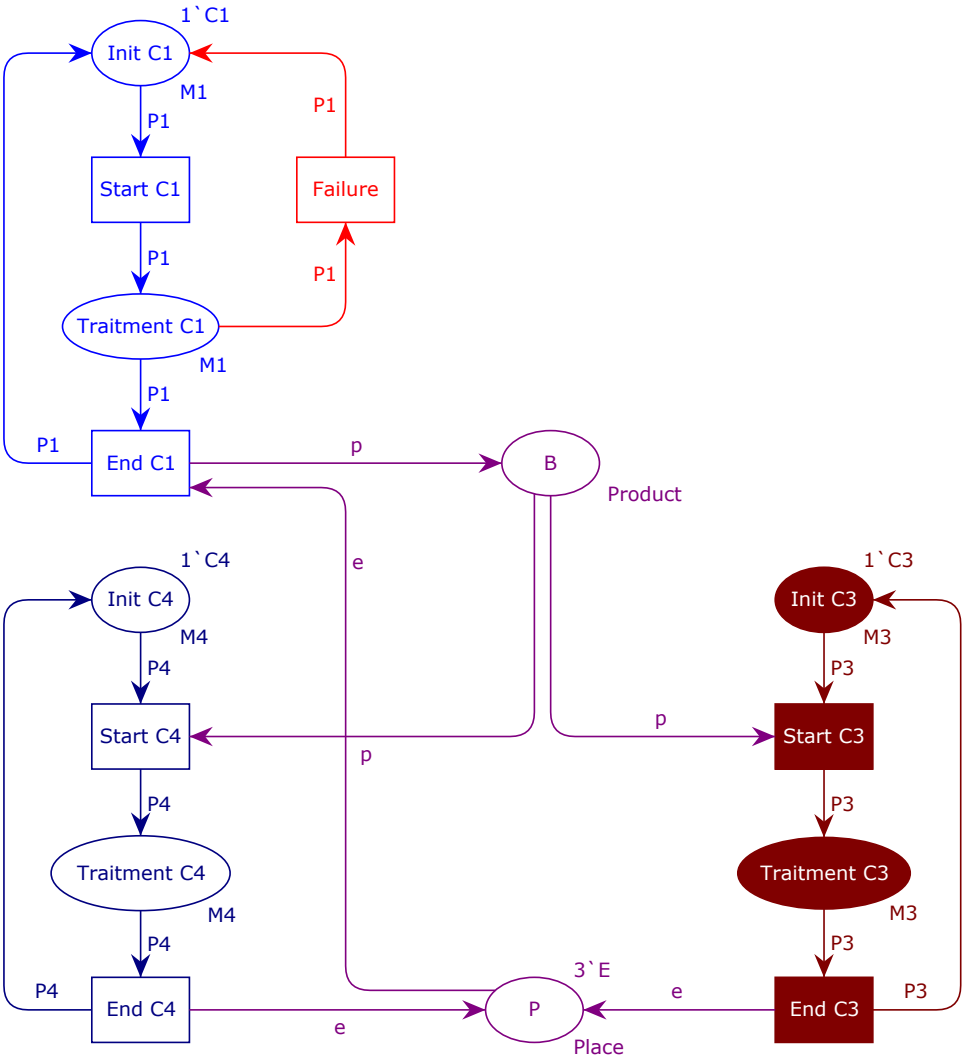



Figure 3.10 – Modèle de *Nominal* étendu

Ainsi l'activation/désactivation des modèles est assuré par l'ajout des éléments suivant :

— La place *Mode Management* caractérisée par le domaine de couleur *mode* et susceptible de contenir un et un seul jeton dont la couleur est associée au mode actuellement actif. Le domaine de couleur de cette place est l'ensemble des couleurs $Col(\bigcup OM_i)$. Initialement *Mode Management* contient la couleur $om_1$ ;

— Définir une variable $M$ sur l'ensemble des couleurs : Cette variable peut s'instancier avec n'importe quelle couleur de $Col(\bigcup OM_i)$ ;

— Associer à chaque transition du $OM_i$ une garde de la forme $M = om_i$ avec $Col(\bigcup OM_i) = om_i$ ;

— Relier la place *Mode Management* à toutes les transitions de tous les modèles par des arcs dont la fonction de couleur est $M$ ;

— Relier les transitions de commutation à la place *Activated mode* par deux arcs. Un

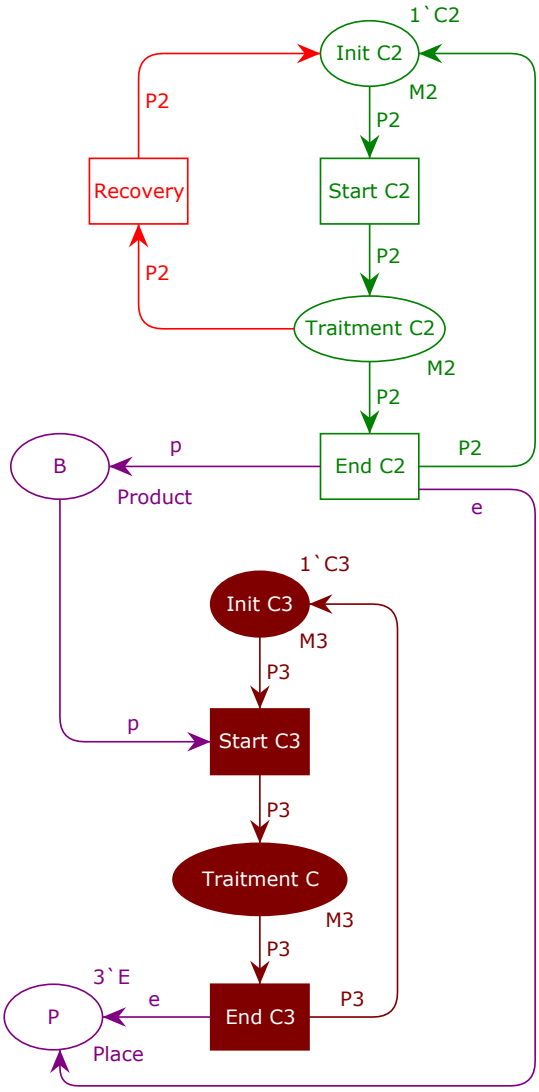



Figure 3.11 – Modèle de *Degraded* étendu

arc qui absorbe le jeton qu'elle contient dont la fonction de couleur est $M$ et un arc qui lui génère un jeton de couleur correspondant au mode à activer.

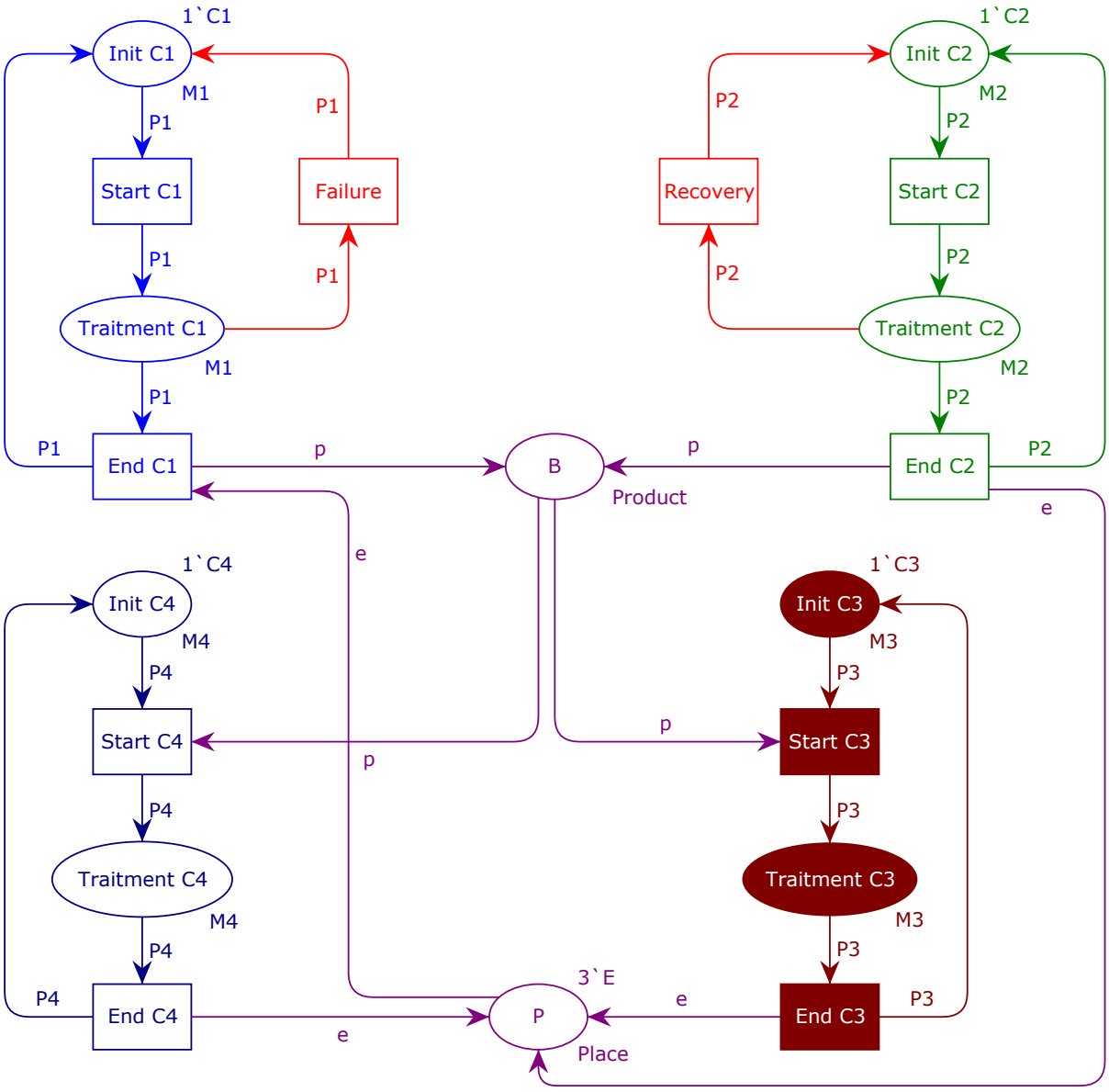



Figure 3.12 – Modèle obtenu par l'algorithme de fusion

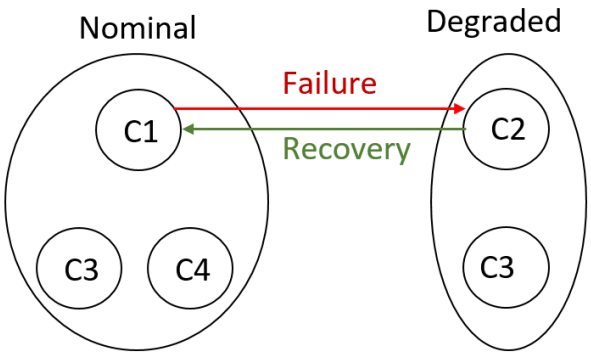



Figure 3.13 – Exemple d'une décomposition multi-modèle

L'Algorithme 4 produit le modèle RdPC final en mettant à jour le modèle global obtenu avec l'Algorithme 3 par l'ajout du mécanisme de commutation.

**Algorithme 4 :** Génération du modèle système

**Entrées :** modèle RdPC $G =< P, T, K, D, W^-, W^+, \phi, M_0 >$ ;
la fonction $target\_om$ l'ensemble $OM$ de modes de fonctionnement ;
l'ensemble des RdPCs $OM_i, (i = 1...|OM|)$ ;
**Output :** un modèle RdPC $G =< P, T, K, D, W^-, W^+, \phi, M_0 >$

$P \leftarrow P \cup \{Mode\ Management\}$;
$K \leftarrow K \cup \{mode\}$;
$\phi'(t) \leftarrow 0$;
$M_0(Mode\ Management) \leftarrow OM_1$;
**pour chaque** *mode de fonctionnement* $OM_i, i \leftarrow 1$**à** $|OM|$ **faire**
  **pour chaque** $t \in T$, **faire**
    $W^-(Mode\ Management, t) \leftarrow M$;
    **si** $target\_mode(t) \leftarrow OM_i$ **alors**
      $W^+(Mode\ Management, t) \leftarrow M$;
    **fin**
    **sinon**
      $W^+(Mode\ Management, t) \leftarrow target\_mode(t)$;
    **fin**
  **fin**
  **pour chaque** $t \in (T_{OM_i} \setminus T_{C^{\Leftrightarrow}_{OM_i}})$ **faire**
    $\phi(t) = \phi_i(t) \wedge (M = OM_i)$ ;
  **fin**
  **pour chaque** $t \in T_{C^{\Leftrightarrow}_{OM_i}}$ **faire**
    $\phi'(t) = \phi'(t) \vee (M = OM_i)$ ;
  **fin**
**fin**
**pour chaque** $t \in T$ **faire**
  $\phi(t) \leftarrow \phi(t) \vee (\phi'(t))$ ;
**fin**

En appliquant à notre exemple directeur, nous obtenons $mode = \{Nominal, Degraded\}$, $Mode\ Management = Nominal$ et le modèle final illustré par la Figure 3.14.

## 3.7 Conclusion

Dans ce chapitre, nous avons présenté une démarche de conception des systèmes complexes basé sur l'approche multi-modèle. Notre démarche est décomposée en étapes et elle

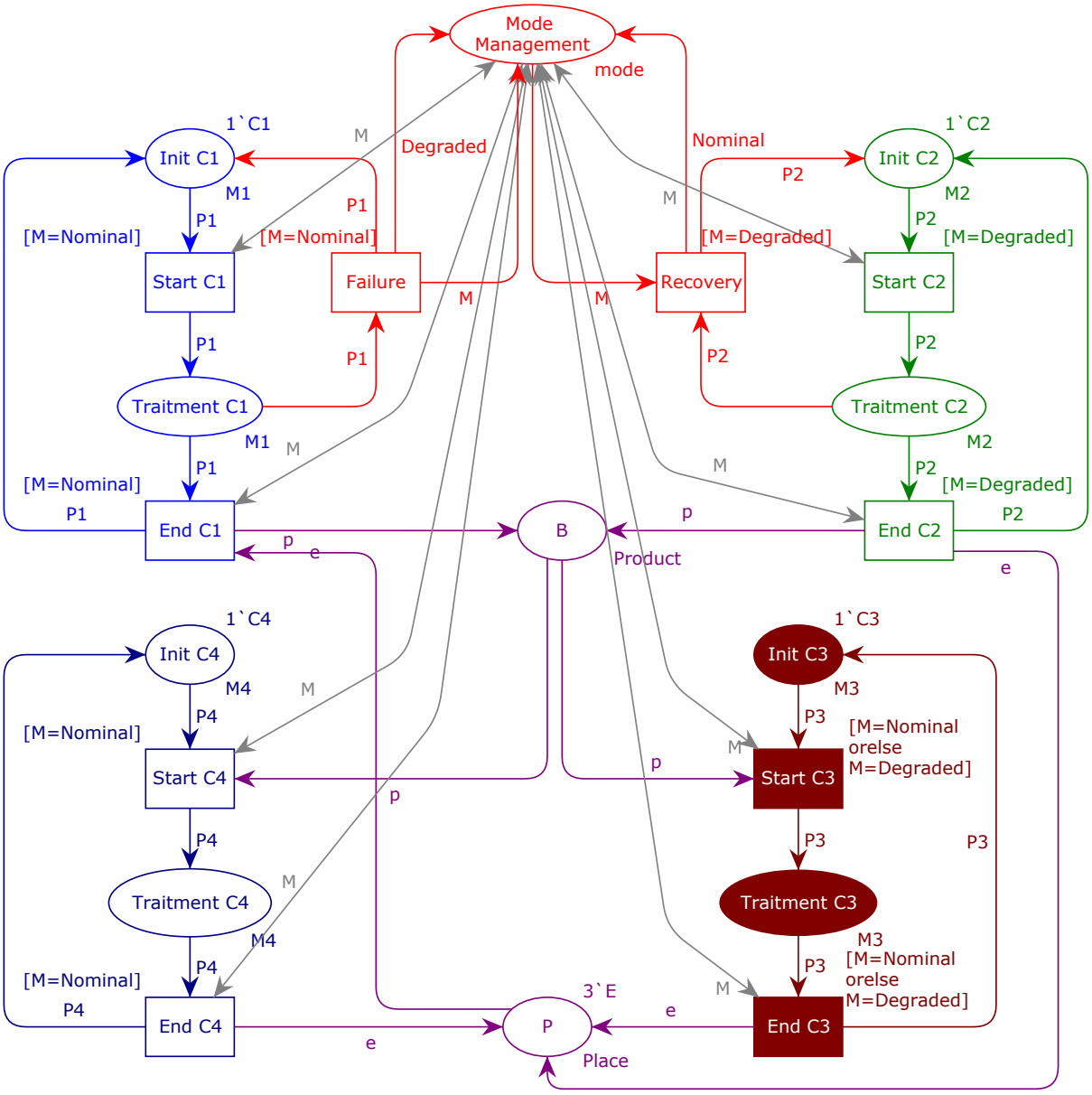



Figure 3.14 – Modèle global du système étudié

structure la spécification du système en plusieurs modes de fonctionnement comprenant chacun un ensemble différent de composants. En appliquant une conception ascendante, la première étape de la démarche est Modèle de composants et elle consiste à modéliser chaque composant dans un modèle RdPC à part. La deuxième étape est Modèle de modes où chaque mode est construit indépendamment des autres à partir des modèles RdPC des composants. La troisième étape est Modèle du système. Elle se concentre sur la conception d'un modèle unique fusionnant tous les modèles de modes sans duplication de composants communs et en intégrant un mécanisme de commutation. Dans le chapitre suivant, nous allons étendre nos travaux pour traiter le problème de conception des SdSs dynamiques.

# Chapitre 4

# Démarche de conception des SdS

## 4.1 Introduction

Le concept de SdS a été proposé dans le contexte des systèmes complexes de grande taille qui travaillent ensemble pour atteindre des objectifs communs. Il offre un point de vue de haut niveau englobant les interactions entre les systèmes constitutifs [Jamshidi 2008]. Le dysfonctionnement d'un système donné peut avoir de graves conséquences sur les performances de l'ensemble du SdS, il est donc important que la conception prenne en compte les exigences de fiabilité, de robustesse et de la sécurité de fonctionnement. Partant de cette nécessité, l'objectif de ce chapitre est de concevoir de manière formelle les SdS tout en maintenant un fonctionnement acceptable malgré les défaillances à travers une démarche hiérarchique de modélisation. Pour faciliter la compréhension de la démarche, nous allons l'illustrer par un exemple manufacturier.

## 4.2 Exemple directeur

Le SdS étudié est un SdS manufacturier inspiré de la littérature. Il comporte deux systèmes $System1$ et $System2$, chacun est composé de trois machines et d'un stock, comme illustré dans la Figure 4.1.

$System1$ fabrique des pièces de type A qui sont par la suite livrées à $System2$ pour être utilisées dans la fabrication des pièces de type B.

$System1$ est constitué des machines $C11$, $C12$ et $C13$ et d'un stock interne $B1$ de capacité 3 alors que $System2$ est constitué des machines $C21$, $C22$ et $C23$ et d'un stock interne $B2$ de capacité 3 aussi. Les stocks amont et aval aux $System1$ et $System2$ ne sont pas considérés. Chacun de ces deux systèmes comporte deux modes de fonctionnement, un mode nominal et un mode dégradé.

Le fonctionnement de $System1$ est le suivant : durant le mode nominal, $Nominal1$,

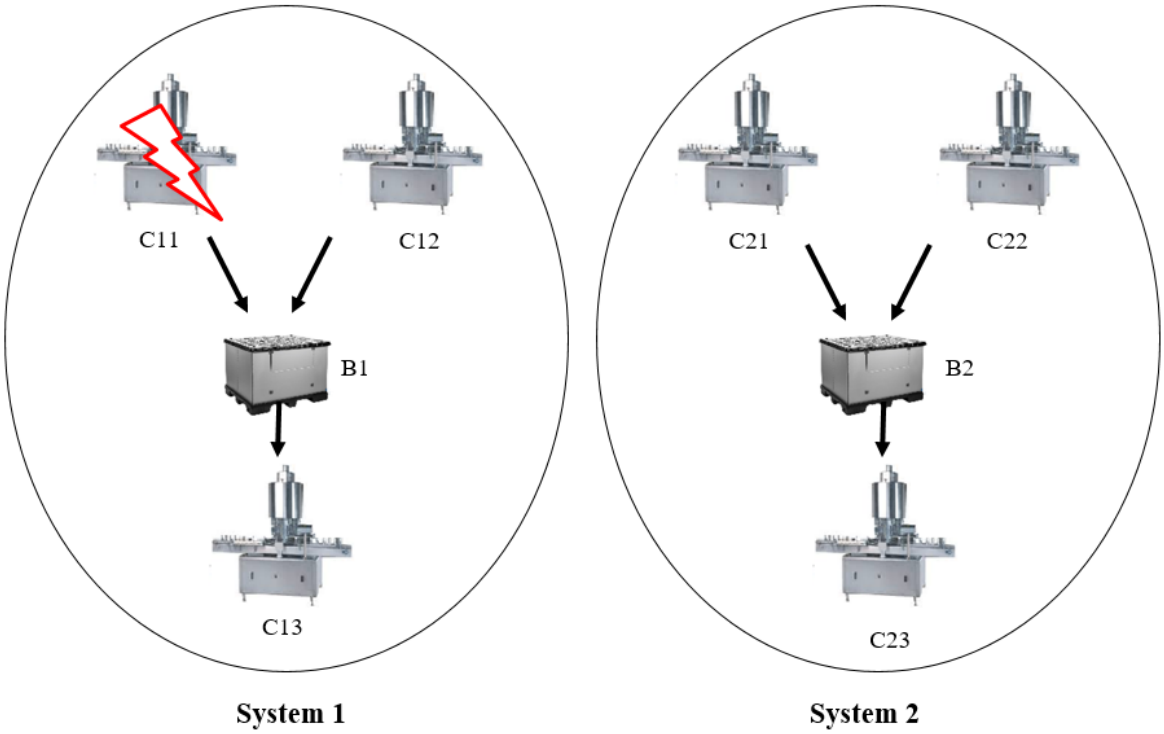



Figure 4.1 – SdS manufacturier.

la machine $C12$ est au repos. La machine $C11$ prend une à une les pièces d'un stock en amont, effectue un traitement puis les dépose dans le stock intermédiaire $B1$. L'arrivée d'une pièce au stock $B1$ l'incrémente de 1. $C13$ prélève une à une les pièces du stock $B1$, réalise un traitement puis les dépose dans un stock en aval pour être transférés au $system2$. Le lancement d'une tâche sur $C13$ décrémente le stock $B1$ de 1.
La machine $C11$ peut tomber en panne durant le déroulement d'une tâche. Cette panne est symbolisée par l'événement *Failure*. L'occurrence de cet événement implique une commutation du $system1$ du mode $Nominal1$ vers un mode dégradé, $Degraded1$, dans lequel la machine $C11$ n'est plus utilisée jusqu'à l'occurrence de l'événement *Recovery* qui implique le retour du système dans le mode nominal.
Le mode $Degraded1$ de $System1$ représente le comportement du $system1$ lorsque la machine $C11$ est en état de panne. Dans ce mode, la machine $C11$ est remplacée par la machine $C13$ ; $C13$ est dite redondante avec $C11$. Cependant, la machine $C13$ est moins rapide que la machine $C11$, ce qui implique la diminution de la production du $System1$. Ceci influence le fonctionnement de $System2$ qui a besoin des pièces produites par $System1$.

Le fonctionnement de $System2$ est le suivant :
$C21$ et $C22$ sont deux machines dont le fonctionnement est similaires. Durant le mode nominal, $Nominal2$, elles récupèrent une à une les pièces d'un stock en amont sans notion de priorité, réalisent un traitement puis les déposent dans un stock intermédiaire $B2$. La fin

de traitement d'une pièce de $C21$ ou de $C22$ incrémente le stock $B2$ d'une unité. Ensuite, la machine $C23$ récupère une à une les pièces du stock $B2$, réalise un traitement puis dépose la pièce traitée dans un stock en aval. Chaque récupération de pièce décrémente le stock $B2$ d'une unité.
Le passage du $System1$ au mode $Degraded1$ influence le fonctionnement de l'ensemble du SdS. En effet, $System2$ n'est plus alimenté en quantités suffisantes et il est obligé de passer en mode dégradé, $Degraded2$, pour adapter sa production en fonction de la disponibilité de la matière première. Dans ce mode, les machines $C21$ et $C23$ continuent à fonctionner alors que la machine $C22$ passe en veille. Lorsque $System1$ est de nouveau au mode $Nominal1$, $System2$ bascule lui aussi au mode $Nominal2$.

## 4.3 Présentation de la démarche

Ce chapitre présente une démarche formelle de conception ascendante des SdSs dynamiques dont la structure est illustrée à la Figure 4.2. Un SdS se compose de plusieurs systèmes, chaque système étant composé de plusieurs modes de fonctionnement, chaque mode de fonctionnement étant également composé de plusieurs composants.

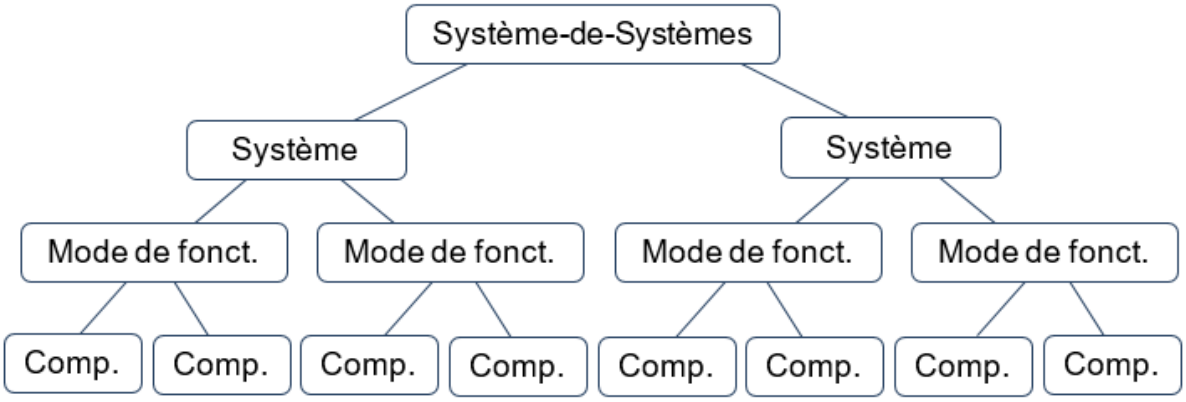



Figure 4.2 – Structure hiérarchique d'un SdS

La démarche proposée est à quatre étapes (voir Figure 4.3). La première étape de cette démarche est «Modèle de composant», dans laquelle les composants sont modélisés par des modèle RdPC distinctes conformément à leurs spécifications. La deuxième étape est «Modèles de modes» et elle consiste à étudier indépendamment et séparément chaque mode de fonctionnement en appliquant de manière conventionnelle la TCS. Grâce à l'utilisation des RdPCH, le modèle de mode est composé de deux couches. La couches inférieure représente les modèles de composants de la première étape et la couche supérieure représente les relations de liaison entre ces modèles de composants. La troisième étape est «Modèle de systèmes». Elle permet de modéliser chaque système avec ses actions de reconfiguration.

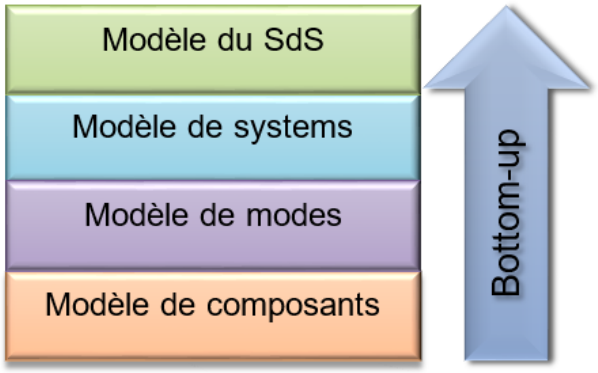



Figure 4.3 – Modèle hiérarchique de conception du SdS

Chaque modèle de système est un modèle RdPCH composée de trois couches dont les deux premiers sont issue de l'étape précédente. La couche supérieure contient des transitions de substitution, chacune étant associée à un sous-modèle de mode. Cette couche intègre un mécanisme de commutation permettant de basculer d'un mode à un autre.
La dernière étape est «Modèle du SdS», dans laquelle les dépendances entre les systèmes d'un SdS sont modélisées en réponse à des reconfigurations internes des systèmes. Le modèle du SdS issu de cette étape est composé de quatre couches dont trois sont construites à l'étape précédente. La couche SdS est la couche supérieure du modèle du SdS. Elle intègre à des transitions de substitution associées chacunes à un modèle de système et un mécanisme de dépendance permettant d'assurer les relations de reconfigurations entre les systèmes.
Les sections suivantes détaillent chaque étape. Mais tout d'abord, introduisons les définitions suivantes :

**Définition 15 (Ensemble de systèmes)**
*L'ensemble des systèmes d'un SdS est appelé* $S = \{S_1, S_2, .., S_{|S|}\}$ *où* $|S| > 1$.

Le SdS étudié possède deux systèmes, $S = \{System1, System2\}$.

Les systèmes composant le SdS sont décrits sous forme de mode de fonctionnement représentant chacun un comportement réduit du comportement global du système selon la définition suivante :

**Définition 16 (Ensemble de modes)**
*L'ensemble des modes de fonctionnement d'un système* $S_i$, *où* $i \in \mathbb{N}$ *et* $i \in \{1..|S|\}$ *est appelé* $OM_i = \{OM_{i,1}, OM_{i,2}, .., OM_{i,|OM_i|}\}$ *où* $|OM_i| > 1$.

Dans le cas de notre exemple, les modes de fonctionnement du système $System1$ sont $OM_1 = \{Nominal1, Degraded1\}$ et ceux de $System2$ sont $OM_2 = \{Nominal2, Degraded2\}$.

Les modes de fonctionnement sont à leurs tour composés d'un ensemble de composants. Cet ensemble n'est pas forcément le même dans chaque mode de fonctionnement.

**Définition 17 (Ensemble de composants d'un système)**
*Soit un système $S_i$, où $i \in \mathbb{N}$ et $i \in \{1..|S|\}$.*
*L'ensemble des composants d'un système $S_i$ est appelé $C_i = \{C_{i,1}, C_{i,2}, .., C_{i,|C_i|}\}$ où $|C_i| > 1$.*

**Définition 18 (Ensemble de composants d'un mode)**
*Soient un système $S_i$, où $i \in \mathbb{N}$ et $i \in \{1..|S|\}$ et $OM_i$ l'ensemble des modes de fonctionnement correspondant.*
*L'ensemble des composants du mode $OM_{i,j}$, où $j \in \mathbb{N}$ et $j \in \{1..|OM_i|\}$, est appelé $C_{OM_{i,j}}$ tel que $C_{OM_{i,j}} \subseteq C_i$.*

Appliquées à notre exemple, le SdS possède six machines et deux stocks. Chaque machine est considérée ici comme un composant. Le stock n'est pas considéré comme un composant mais plutôt comme une spécification système. Cela correspond à un choix classique de contrainte TCS [Ramadge et Wonham 1989] car le stock n'est pas un événement générateur qui lui soit propre. Nous obtenons alors :

$C_1 = \{C11, C12, C13\}$, $C_2 = \{C21, C22, C23\}$,
$C_{Nominal1} = \{C11, C13\}$, $C_{Nominal2} = \{C21, C22, C23\}$,
$C_{Degraded1} = \{C12, C13\}$, $C_{Degraded2} = \{C21, C23\}$.

**Définition 19 (Ensemble des modes)**
*Soit $S$ l'ensemble des systèmes d'un SdS.*
*L'ensemble de tous les modes de fonctionnement d'un SdS $S$ est appelé $OM = \{OM_1, OM_2, .., OM_{|OM|}\}$ où $|OM| > |S|$.*

**Définition 20 (Ensemble des composants)**
*Soit $S$ l'ensemble des systèmes d'un SdS.*
*L'ensemble de tous les composants de $|S|$ est appelé $C = \{C_1, C_2, .., C_{|C|}\}$ où $|C| > |OM|$.*

Nous obtenons alors, par application à notre exemple :

$OM = \{Nominal1, Degraded1, Nominal2, Degraded2\}$,
$C = \{C_1, C_2\} = \{C11, C12, C13, C21, C22, C23\}$.

## 4.4 Modèle de composants

Chaque système est composé de plusieurs composants dont le comportement est identique quel que soit le mode de fonctionnement. Certains d'eux peuvent déclencher des

évènements exceptionnels provoquant une reconfiguration tel que les évènements de défaillance ou de récupération. Cette étape de notre approche vise à modéliser tous les composants par des modèles RdPC séparées tout en précisant les évènements internes et de commutations et les ports de communications qui relie les composants les uns aux autres.

**Définition 21 (Modèles de composant)**
*Soient un système* $S_i \in S$, *et* $C_i$ *l'ensemble des composants associés. Un composant* $C_{i,j}$ *où* $i \in \{1..|S|\}$ *et* $j \in \{1..|C_i|\}$ *est modélisé par un RdPC* $< P_{C_{i,j}}, T_{C_{i,j}}, K_{C_{i,j}}, W^-_{C_{i,j}}, W^+_{C_{i,j}}, \phi_{C_{i,j}}, M_{0,C_{i,j}} >$ *tel que :*

$P_{C_{i,j}} = P^{\circlearrowleft}_{C_{i,j}} \cup P^{\leftrightarrows}_{C_{i,j}}$ *avec* $P^{\circlearrowleft}_{C_{i,j}} \cap P^{\leftrightarrows}_{C_{i,j}} = \emptyset$. $P^{\circlearrowleft}_{C_{i,j}}$ *et* $P^{\leftrightarrows}_{C_{i,j}}$ *sont, respectivement, des places internes et des places de fusion (communication) du composant* $C_{i,j}$.

$T_{C_{i,j}} = T^{\circlearrowleft}_{C_{i,j}} \cup T^{\leftrightarrows}_{C_{i,j}}$ *avec* $T^{\circlearrowleft}_{C_{i,j}} \cap T^{\leftrightarrows}_{C_{i,j}} = \emptyset$. $T^{\circlearrowleft}_{C_{i,j}}$ *et* $T^{\leftrightarrows}_{C_{i,j}}$ *sont, respectivement, les transitions internes et les transitions de commutation du composant* $C_{i,j}$.

$W^-_{C_{i,j}} = W^-_{C_{i,j_{\circlearrowleft}}} \cup W^-_{C_{i,j_{\leftrightarrows}}}$ *avec* $W^-_{C_{i,j_{\circlearrowleft}}} \cap W^-_{C_{i,j_{\leftrightarrows}}} = \emptyset$. $W^-_{C_{i,j_{\circlearrowleft}}}$ *et* $W^-_{C_{i,j_{\leftrightarrows}}}$ *sont, respectivement, des arcs internes et des arcs de commutation (liés à des transitions de commutation) du composant* $C_{i,j}$.

$W^+_{C_{i,j}} = W^+_{C_{i,j_{\circlearrowleft}}} \cup W^+_{C_{i,j_{\leftrightarrows}}}$ *avec* $T_{C_{i,j_{\circlearrowleft}}} \cap W^+_{C_{i,j_{\leftrightarrows}}} = \emptyset$. $W^+_{C_{i,j_{\circlearrowleft}}}$ *et* $W^+_{C_{i,j_{\leftrightarrows}}}$ *sont, respectivement, des arcs internes et des arcs de commutation du composant* $C_{i,j}$.

Les machines à états des Figures 4.4 à 4.9 représentent les modèles de composants de l'exemple de la Figure 4.1. Pour $system1$, $C11$, $C12$ et $C13$ sont modélisés, respectivement, par les Figures 4.4, 4.5 et 4.6 et pour $system2$, $C21$, $C22$ et $C23$ sont représentés, respectivement, par les Figures 4.7, 4.8 et 4.9.

En appliquant la Définition 21, nous obtenons alors pour tout composant $Cij$, $i \in 1, 2$ et $j \in 1, 2, 3$ :

$P^{\circlearrowleft}_{Cij} = \{Init\ Cij,\ Treatment\ Cij\}$,

$P^{\leftrightarrows}_{Cij} = \{Bi, Pi\}$, $Bi$ et $Pi$ représentent respectivement le stock et sa capacité,

$T^{\circlearrowleft}_{Cij} = \{Start\ Cij,\ End\ Cij\}$,

$W^-_{Cij}$, $W^+_{Cij}$ sont étiquetés par $Pij$ défini par la classe de couleur $Mij$, par $p$ défini par la classe de couleur $Product$, ou par $e$ défini par la classe de couleur $Place$,

$M_{0,Cij}(Init\ Cij) = Pij$, $M_{0,Cij}(Pi) = 3`E$.

Pour le modèle RdPC du $C11$, nous avons en plus :

$T^{\leftrightarrows}_{C11} = \{Failure,\ Recovery\}$.

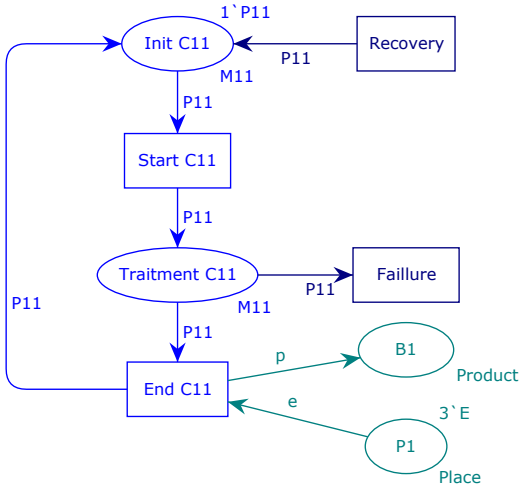



Figure 4.4 – Modèle de $C11$

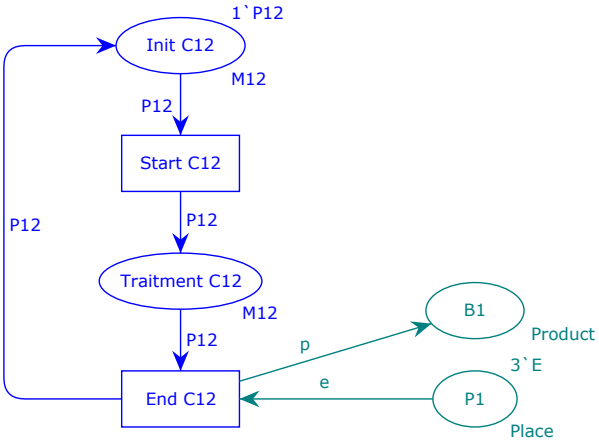



Figure 4.5 – Modèle de $C12$

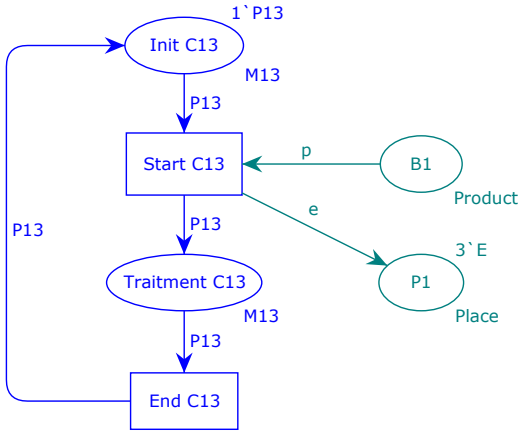



Figure 4.6 – Modèle de $C13$

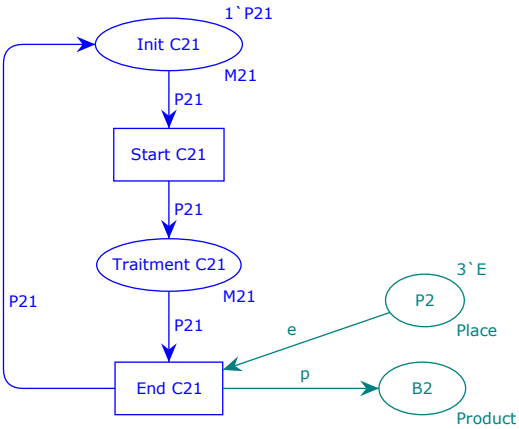



Figure 4.7 – Modèle de $C21$

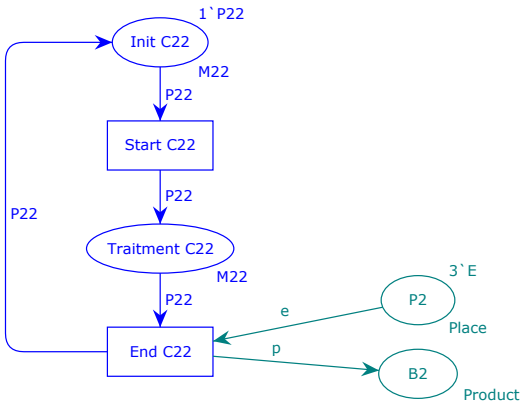



Figure 4.8 – Modèle de $C22$

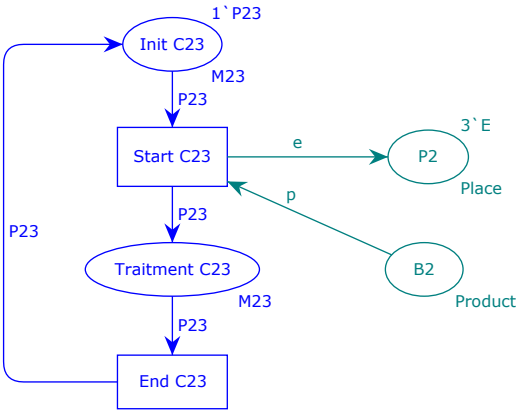



Figure 4.9 – Modèle de $C23$

## 4.5 Modèle de modes

L'objectif de cette section est d'assurer le comportement interne des modes de fonctionnement et qu'ils sont bien construits conformément aux exigences. La commutation d'un mode de fonctionnement à un autre entraîne un changement de la structure du modèle de processus en engageant de nouveaux composants et en libérant d'autres. Par conséquent, chaque mode doit définir l'ensemble des composants nécessaires pour effectuer ses tâches conformément à la spécification. Il doit également faire la distinction entre ses propres composants, utilisés dans un seul mode de fonctionnement, et ses composants communs, utilisés dans au moins deux modes de fonctionnement du même système.

**Définition 22 (Composant propre & composant commun)**
*Soient $S_i$ un système, $OM_i$ et $C_i$ respectivement l'ensemble de modes de fonctionnement et l'ensemble des composants associés et, pour tout mode $OM_{i,j} \in OM_i$, $C_{OM_{i,j}}$ l'ensemble des composant du mode $OM_{i,j}$.*

*Un composant $C_{i,k} \in C_i$, est appelé propre à $OM_{i,j}$ si et seulement si $C_{i,k} \in C_{OM_{i,j}}$ et $\forall OM_{i,p} \in OM_i$, tel que $p \neq j$, $C_{i,k} \notin C_{OM_{i,p}}$.*

*Un composant $C_{i,k} \in C_i$, est appelé commun à $OM_{i,j}$ et $OM_{i,p}$ si et seulement si $C_{i,k} \in C_{OM_{i,j}} \cap C_{OM_{i,p}}$ avec $j \neq p$.*

*Notons $C_{OM_{i,j}} = C^{\circlearrowleft}_{OM_{i,j}} \cup C^{\Leftrightarrow}_{OM_{i,j}}$ où :*

- *$C^{\circlearrowleft}_{OM_{i,j}}$ l'ensemble des composants propres de $OM_{i,j}$ ;*
- *$C^{\Leftrightarrow}_{OM_{i,j}}$ l'ensemble des composants communs de $OM_{i,j}$.*

Dans cette étape, nous visons à modéliser les modes de fonctionnement en appliquant l'approche multi-modèle, qui consiste à concevoir un modèle pour chaque mode de fonctionnement via l'utilisation des RdPCHs ascendants. Un réseau hiérarchique ascendant implique de créer les parties les plus détaillées du modèle, appelées aussi pages. Ensuite ces pages sont définies en tant que sous-pages pour les transitions de substitution de la super-page. En appliquant aux modèles de modes, chaque modèle est composée de deux couches. Une couche modes contenant les super-pages des transitions de substitution dont chacune d'elles est associée à une sous-page représentant un modèle du composant. L'ensemble de ces sous-pages est appelé couche composants. La couche modes et la couche composants sont reliées par des places équivalentes sur les deux couches qui sont les places de communication.
Concernant les composants communs, les transitions de substitution font référence à la même sous-page. En conséquence, de multiple instances de la sous-page du composant concernée sont créées. Cependant, le marquage de chaque instance d'une sous-page peut être complètement indépendant des marquages d'autres instances de la même sous-page.

Alors et afin de maintenir le même statut des différentes instances, chaque place des composants communs est fusionnée avec ses correspondants des autres instances. Ainsi, toute modification d'état apportée à une instance d'un composant s'applique également à toutes les autres instances et nous conservons par conséquent le comportement des composants communs lors de la commutation d'un mode à un autre.

À cette étape, une décomposition modale s'avère nécessaire pour permettre de fixer pour chaque mode de fonctionnement, ses composants associés. Appliquées à notre exemple, la Figure 4.10 illustre la composition modale de l'exemple de la Figure 4.1. Nous pouvons

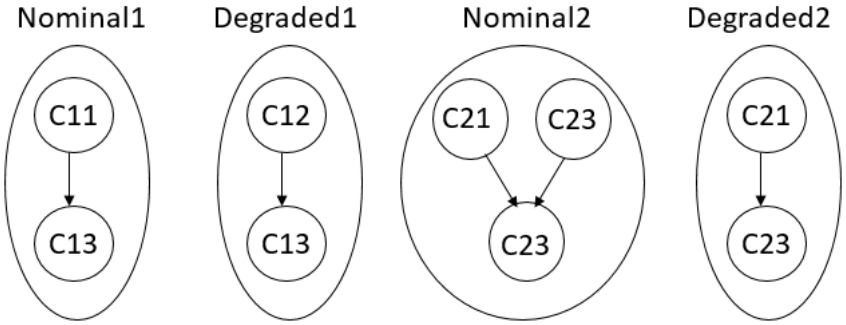



Figure 4.10 – Exemple d'une décomposition modale d'un SdS

remarquer, pour $System1$, que $C_{13}$ est engagé dans les deux modes de fonctionnement, alors que $C_{11}$ contribue à la production en mode $Nominal1$ uniquement et $C_{12}$ intervient lorsqu'une commutation au mode $Degraded1$ est effectuée. Pour $System2$, nous remarquons que toutes les machines sont engagées dans le mode $Nominal2$, mais uniquement, $C21$ et $C23$ sont engagés que dans le mode $Degraded2$. Par conséquent, nous obtenons :

$$C^{\circlearrowleft}_{OM_{Nominal1}} = C11, \qquad C^{\circlearrowleft}_{OM_{Nominal2}} = C22,$$
$$C^{\circlearrowleft}_{OM_{Degraded1}} = C12, \qquad C^{\circlearrowleft}_{OM_{Degraded2}} = \emptyset,$$
$$C^{\Leftrightarrow}_{Nominal1} = C^{\Leftrightarrow}_{Degraded1} = C13, \qquad C^{\Leftrightarrow}_{Nominal2} = C^{\Leftrightarrow}_{Degraded2} = C21, C23.$$

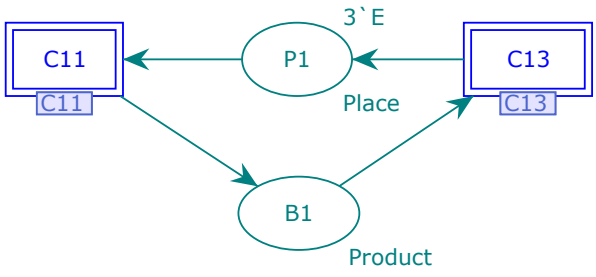
Maintenant, nous sommes en mesure de créer les modèles de modes. L'Algorithme 5 présente les étapes de création des modèles de modes à partir de la spécification des modes de fonctionnement et des modèles de composants déjà construits.
Pour des raisons de simplicité, nous confondons la notation $OM_{i,j}$ d'une identité de mode et son RdPCH associé et la notation $C_{i,j}$ d'une identité de composant et son RdPCH associé aussi.

Appliquées à notre exemple, l'Algorithme 5 nous donnent les Figures 4.11, 4.12, 4.13 et 4.14 qui illustrent les modèles RdPCH des différents modes. Sur ces figures, chaque composant est représenté dans le mode dans lequel il est utilisé.

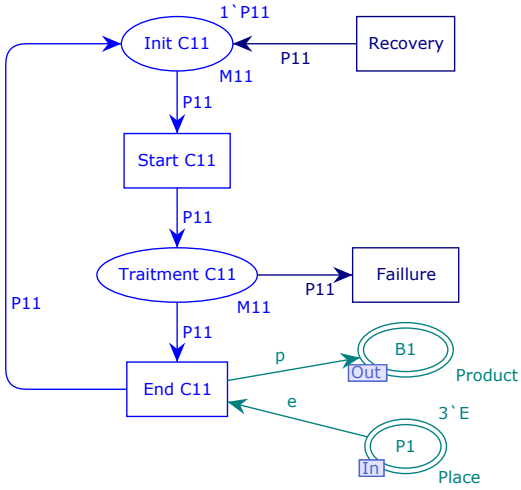
**Algorithme 5 :** Génération des modèles de modes

**Entrées :** l'ensemble $S$ des systèmes ;
l'ensemble des modes $OM_i$, $(i = 1..|S|)$ ;
l'ensemble $C$ des composants ;
l'ensemble $C_{OM_{i,j}}$ des composants $(i = 1..|S|, j = 1..|OM_i|)$ ;
l'ensemble des modèles RdPCs des composants $C_{i,k}$ $(i = 1..|S|, K = 1..|C|)$ ;
**Output :** l'ensemble des modèles RdPCH des modes $OM_{i,j}$

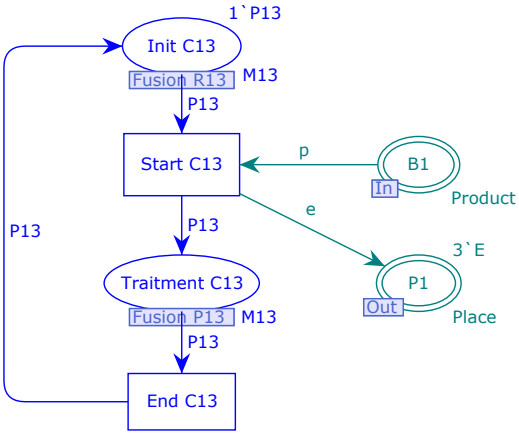
**pour chaque** *systeme* $S_i$, $i \leftarrow 1$**à** $|S|$ **faire**
  **pour chaque** *mode* $OM_{i,j}$, $j \leftarrow 1$**à** $|OM_i|$ **faire**
    Créer une page $OM_{i,j}$;
    **pour chaque** *composant* $C_{i,k} \in C_{OM_{i,j}}$ **faire**
      Ajouter une transition de substitution, $T^{\Updownarrow}_{i,j,k}$, dans $OM_{i,j}$ associée à $C_{i,k}$ ;
      **pour chaque** $P^{C_{i,k}} \in P^{\leftrightarrows}_{C_{i,k}}$ **faire**
        Ajouter $P^{C_{i,k}}$ dans $OM_{i,j}$;
        Ajouter un port-type tag dans $C_{i,k}$;
        Ajouter un arc connectant $P^{C_{i,j}}$ à $T^{\Updownarrow}_{i,j,k}$;
      **fin**
      **si** $C_{i,k} \in C^{\Leftrightarrow}_{OM_{i,j}}$ **alors**
        **pour chaque** $OM_{i,p}$, $p \leftarrow 1$**à** $j$ **faire**
          **si** $C_{i,k} \in C^{\Leftrightarrow}_{M_{i,p}}$ **alors**
            Fusionner $P^{C_{i,k}}_{\circlearrowleft}$ dans les instances de $OM_{i,j}$ et $OM_{i,p}$ ;
          **fin**
        **fin**
      **fin**
    **fin**
  **fin**
**fin**




(a) Super-page *Nominal*1

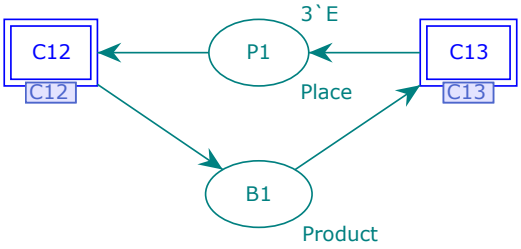



(b) Sous-page $C11$

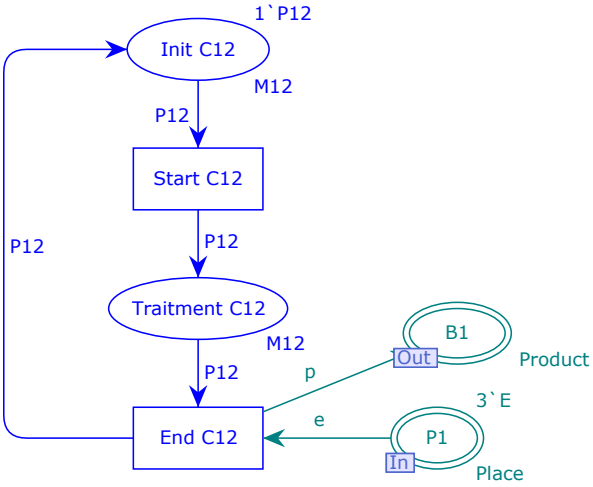



(c) Sous-page $C13$

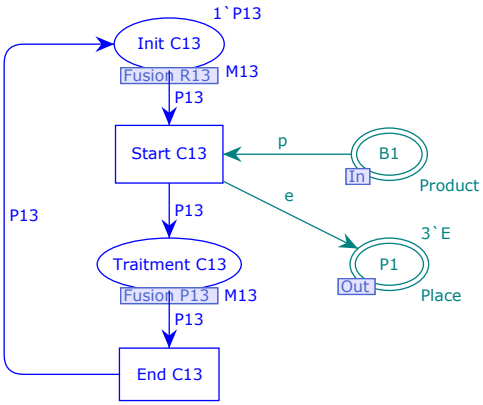
Figure 4.11 – Modèle de *Nominal*1




(a) Super-page *Degraded*1

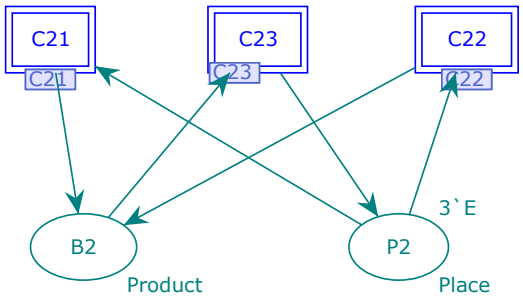



(b) Sous-page $C12$

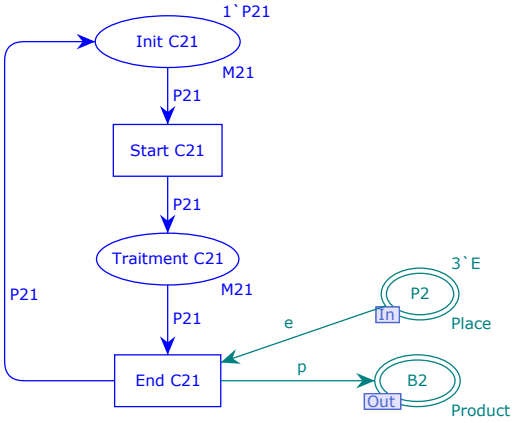



(c) Sous-page $C13$

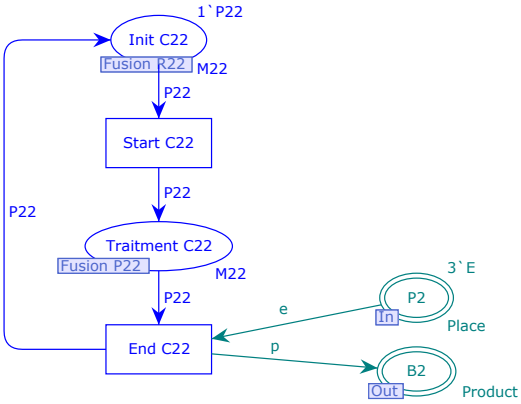
Figure 4.12 – Modèle de *Degraded*1



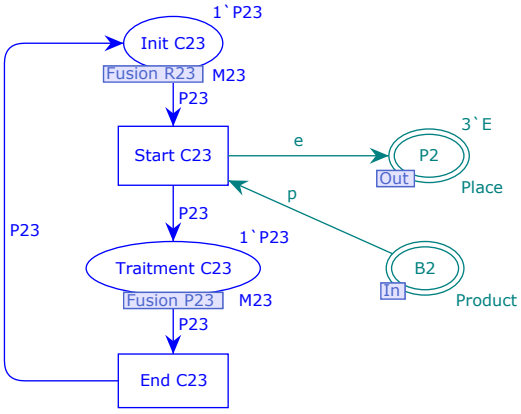

(a) Super-page *Nominal*2




(b) Sous-page *C*21

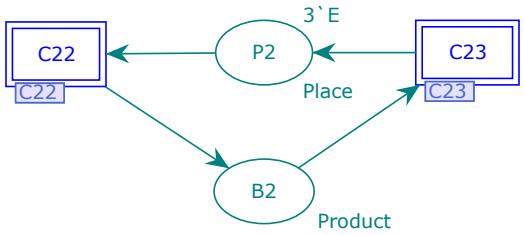



(c) Sous-page *C*22

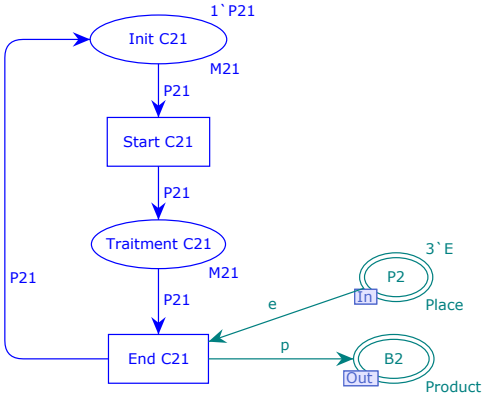



(d) Sous-page *C*23

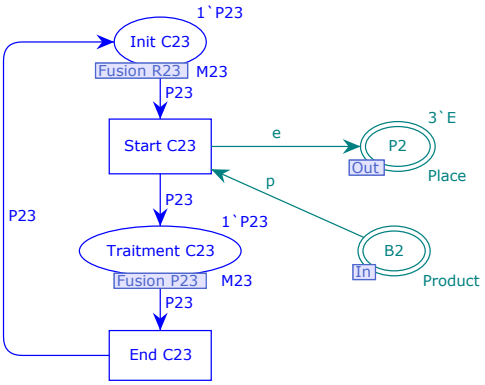
Figure 4.13 – Modèle de *Nominal*2




(a) Super-page *Degraded*2




(b) Sous-page *C*21




(c) Sous-page *C*23

Figure 4.14 – Modèle de *Degraded*2

Néanmoins, ces modèles de mode ne sont pas interconnectés. C'est le but principal de la section suivante.

## 4.6 Modèles de systèmes

Cette section se concentre sur la conception de tous les systèmes composant le SdS. Chaque système est développé séparément des autres en prenant en compte les comportements pouvant conduire à une commutation entre les modes de fonctionnement. La démarche proposée permet de modéliser le comportement global de chaque système par un modèle RdPCH. Chaque modèle du système est composé de trois couches. Les deux premières couches sont la couche composants et la couche modes issues de l'étape Modèle de modes. La troisième couche représente la super-page contenant les transitions de substitution associées aux pages de la couche mode et le mécanisme de commutation permettant au système d'activer ou désactiver les modèles de modes de fonctionnement.

Premièrement, une décomposition multi-modèle de chaque système est nécessaire où les éventuelles commutations entre les modes sont modélisées et vérifiées par rapport à la spécification.

La Figure 4.15 illustre la décomposition multi-modèle des systèmes de l'exemple directeur. Nous avons deux modes par système : *Nomimal*1 et *Degraded*1 pour *System*1 et

$Nomimal2$ et $Degraded2$ pour $System2$. Au niveau de $System1$, il est possible de basculer du mode $Nominal1$ au mode $degraded1$ au moyen de l'événement de commutateur *Failure* généré en cas de panne du composant $C11$. Le retour au mode $Nominal1$ s'effectue lorsque $C12$ génère l'événement *Recovery*. $System2$ peut aussi basculer du mode $Nominal2$ au mode $Degraded2$ et inversement mais l'évènement déclencheur n'est pas interne au système. En effet, $System2$ commute de mode pour s'adapter aux reconfigurations réalisées au sein du SdS. Ces évènements externes seront traités dans la section suivante.

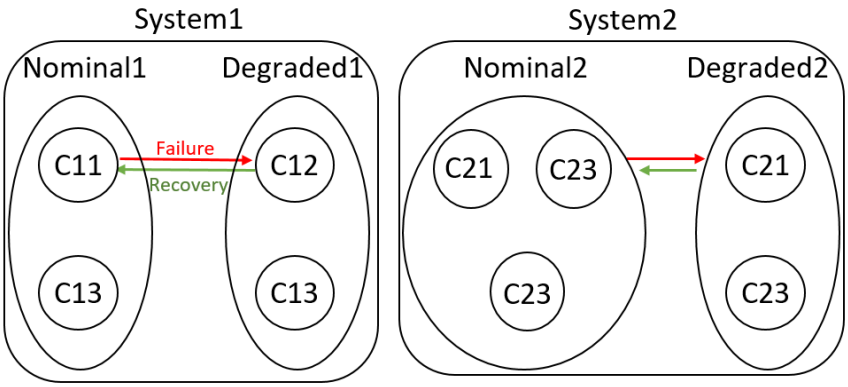



Figure 4.15 – Décomposition multi-modèle de l'exemple du sds

Deuxièmement, un mécanisme de commutation est à intégrer dans la couche système de chaque système.

Les transitions $T_{\leftrightarrows}^{C_{i,k}}$ où $i \in \{1..|S|\}$ et $k \in \{1..|C_i|\}$ sont les transitions de commutations. Franchir de telles transitions doit désactiver toutes les transitions du mode courant et permettre l'activation des transitions du mode destination. Pour distinguer ce type de transitions, nous définissons une application notée $target\_mode$ dont le rôle est d'associer à chaque transition son mode de destination.

**Définition 23 (Mécanisme de commutation)**
*Soient $S_i$ un système et $OM_i$ l'ensemble des modes de fonctionnement associé. Soient $OM_{i,j} \in OM_i$, $T_{OM_{i,j}}$ l'ensemble des transitions du modes $OM_{i,j}$ et $T_{OM_{i,j}}^{\leftrightarrows} \subset T_{OM_{i,j}}$ l'ensemble des transitions de commutation associées.*
*$target\_mode : T_{OM_{i,j}}^{\leftrightarrows} \rightarrow OM_i$ est une application tel que $target\_mode(t)$ indique le mode de fonctionnement actif après le déclenchement de $t, \forall t \in T_{OM_{i,j}}^{\leftrightarrows}$.*

De plus, de nouveaux éléments appartenant au mécanisme de commutation doivent être ajoutés à chaque système $S_i$, ces éléments sont détaillés dans le Chapitre 3 à la sous-section 3.6.3 intitulé «Intégration du mécanisme de commutation».

L'Algorithme 6 suivant décrit les étapes nécessaires à la création d'un RdPCH permet-

tant la gestion des modes de fonctionnement de chaque $S_i$.

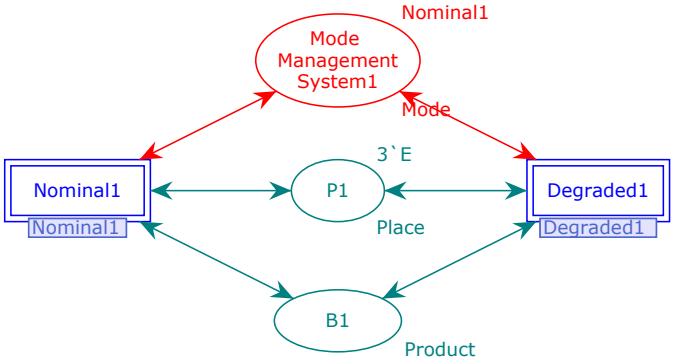
**Algorithme 6 :** Génération des modèles de systèmes

**Entrées :** l'ensemble $S$ de systèmes ;
l'ensemble $OM$ de modes ;
l'ensemble de RdPCH $OM_{i,j}$ $(i = 1..|S|, j = 1..|OM_i|)$ ;
l'ensemble $C$ des composants ;
**Output :** l'ensemble de HRdPC $S_i$ des systèmes

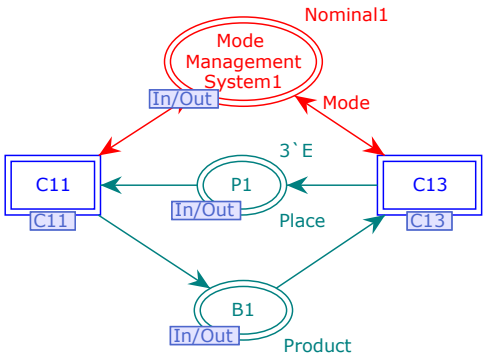
**pour chaque** *système* $S_i$, $i \leftarrow 1$**à** $|S|$ **faire**
  Créer page $S_i$;
  $K = K \cup Mode$;
  Ajouter une place $Mode\ Management\ S_i$ ;
  $M_0(Mode\ Management\ S_i) = OM_{i,1}$;
  **pour chaque** *mode* $OM_{i,j}$, $j \leftarrow 1$**à** $|OM_i|$ **faire**
    Ajouter une transition de substitution, $T^{\Updownarrow}_{i,j}$, associée à $OM_{i,j}$ dans $S_i$;
    Ajouter des arcs connectant $Mode\ Management\ S_i$ à $T^{\Updownarrow}_{i,j}$ ;
    Ajouter une place $Mode\ Management\ S_i$ dans $OM_{i,j}$;
    Ajouter des arcs connectant $Mode\ Management\ S_i$ à $T^{\circlearrowleft}_{C_{i,k}}$ dans $C_{i,k}$;
    Ajouter les «port-type tag» sur $OM_{i,j}$;
    **pour chaque** $C_{i,k}$, $k \leftarrow 1$**à** $|C_i|$ **faire**
      Ajouter une place $Mode\ Management\ S_i$ dans $C_{i,k}$;
      **pour chaque** $t \in T_{C_{i,k}}$ **faire**
        $W^{-}_{C_{i,t}}(Mode\ Management\ S_i, t) = M$;
        **si** $target\_mode(t) = OM_{i,j}$ **alors**
          $W^{+}_{C_{i,t}}(Mode\ Management\ S_i, t) = M$
        **sinon**
          $W^{+}_{C_{i,t}}(Mode\ Management\ S_i, t) = target\_mode(t)$
        **fin**
      **fin**
    **fin**
    **pour chaque** $t \in T^{\circlearrowleft}_{C_{i,k}}$ **faire**
      $\phi(t) = \phi(t) \wedge [M = OM_{i,j}]$
    **fin**
  **fin**
**fin**

En appliquant l'algorithme 6 à notre exemple directeur, les modèles RdPCH obtenus des deux systèmes sont représentés aux Figures 4.16 et 4.16. Les couches sont réparties comme suit : les modèles de la couche composants sont illustrés par les Figures 4.16(d),

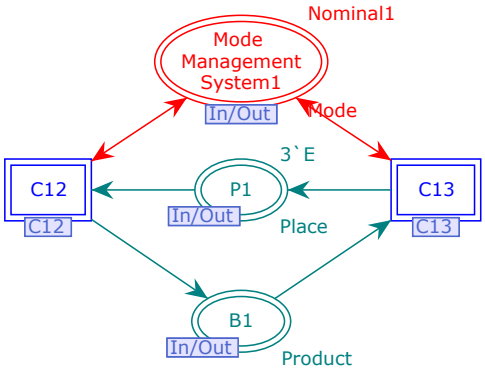
4.16(e), 4.16(f), 4.17(d), 4.17(e) et 4.17(f) ; les modèles de la couches modes sont illustrés par les Figures 4.16(b), 4.16(c), 4.17(b) et 4.17(c) ; et la couche système de RdPCH obtenues est représentée dans les Figures 4.16(a) et 4.17(a).



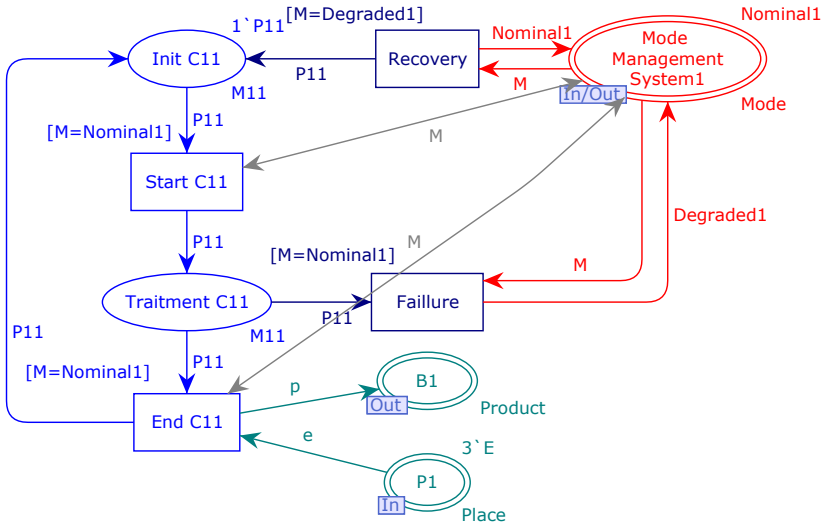

(a) Super-page *System*1



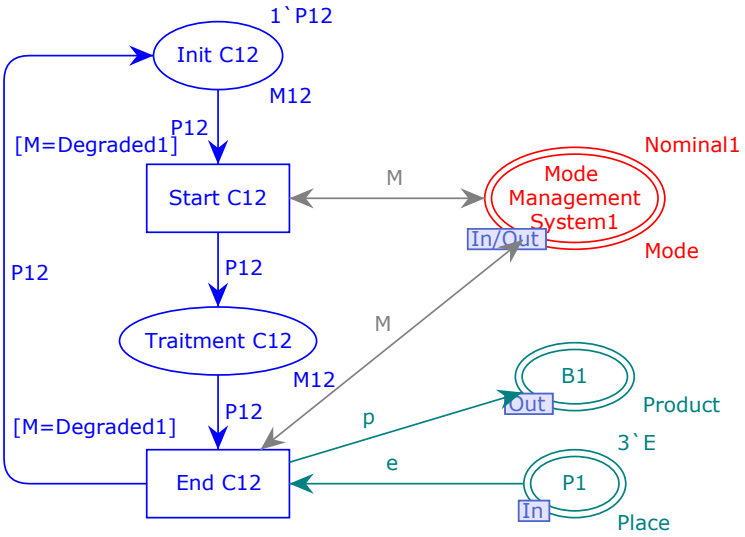

(b) Sous-page *Nominal*1



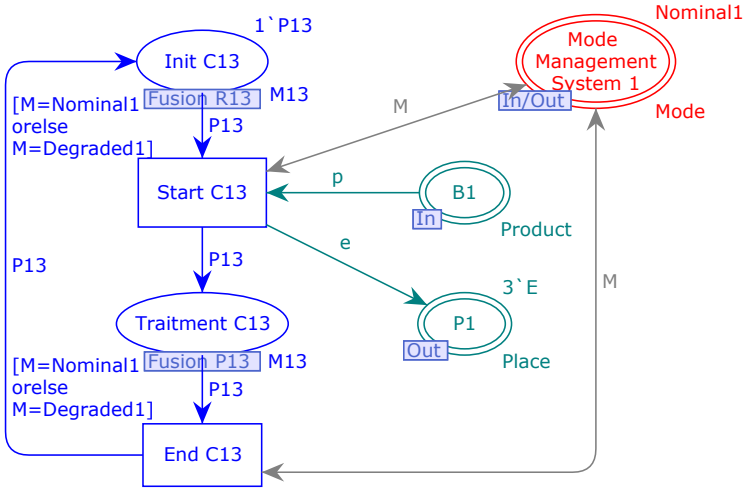

(c) Sous-page *Degraded*1




(d) Sous-page *C*11

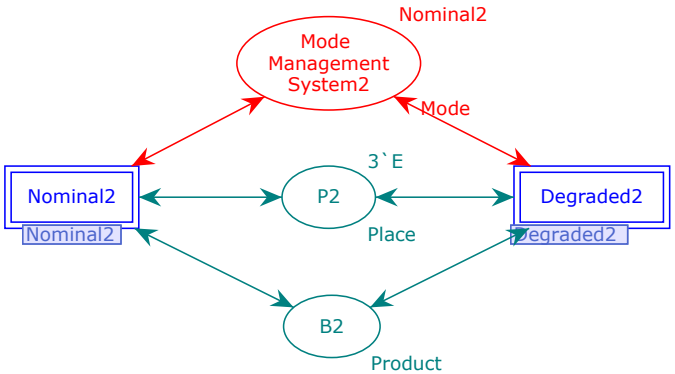



(e) Sous-page $C12$

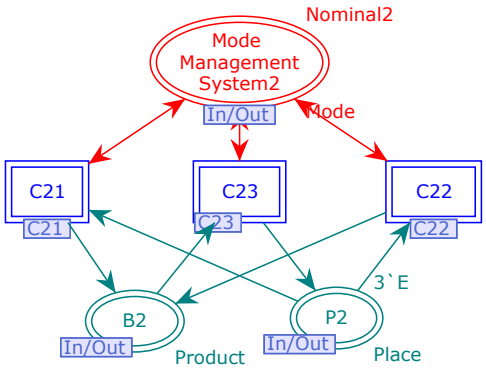



(f) Sous-page $C13$

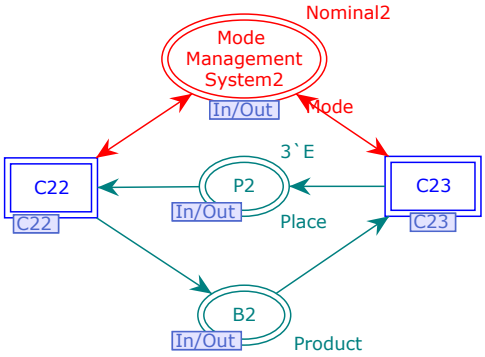
Figure 4.16 – Modèle de $System1$



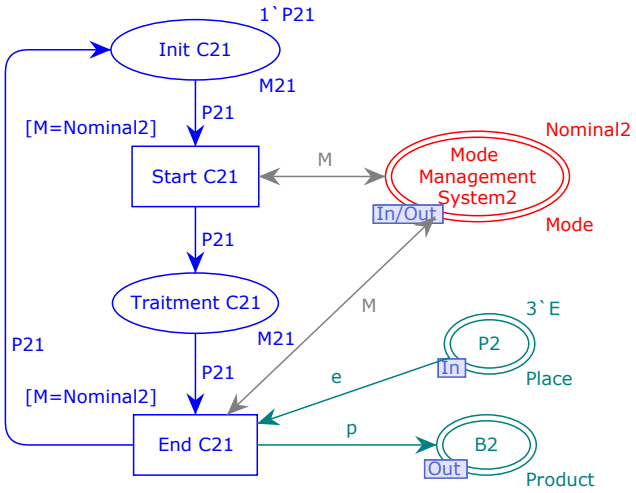

(a) Super-page de $System2$



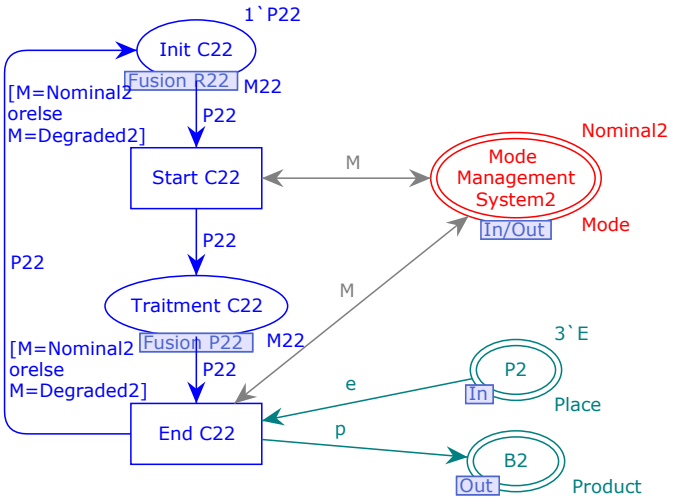

(b) Sous-page *Nominal*2



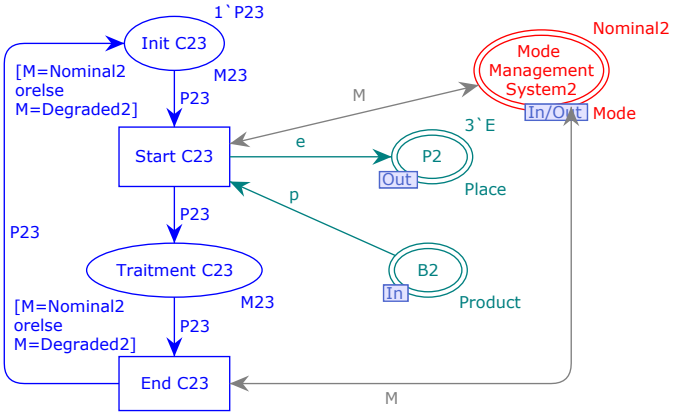

(c) Sous-page *Degraded*2




(d) Sous-page $C21$




(e) Sous-page $C22$




(f) Sous-page $C23$

Figure 4.16 – Modèle de $System2$

## 4.7 Modèle du SdS

Dans cette section, nous créons la couche supérieure du réseau RdPHC dans lequel nous garantissons un comportement cohérent de l'ensemble du modèle malgré les défaillances. En effet, nous visons à maintenir un fonctionnement acceptable du SdS lorsqu'un système bascule en mode dégradé. Nous développons donc l'idée suivante : si un système passe en mode dégradé, un sous-ensemble de systèmes sera affecté et devra se mettre en veille ou basculer vers d'autres modes dégradés dans lesquels un fonctionnement minimal est garanti sous une politique de contrôle révisée. La première étape de cette section est la décomposition système dans laquelle une étude des dépendances entre les systèmes en fonction de leurs commutations de modes est réalisée et vérifiée par rapport à la spécification. La Figure 4.17 représente la décomposition système de l'exemple directeur : un basculement de $System1$ vers le mode dégradé impose à $System2$ de commuter aussi au mode dégradé et vice-versa. Dans cette couche supérieure, un mécanisme de dépendance

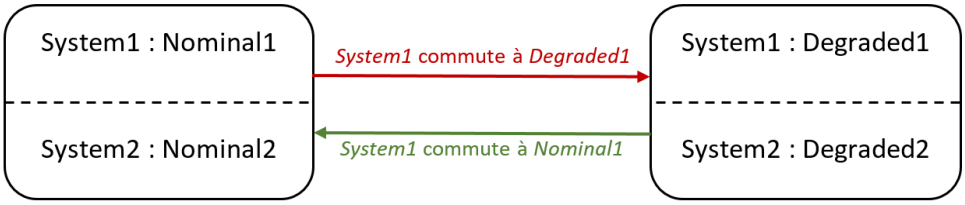



Figure 4.17 – Décomposition système

est intégré. Il est assuré par un nouvel ensemble de transitions de priorité élevée, appelé ensemble de transitions de dépendance, et noté $T_{\rightsquigarrow}$. Chaque transition de dépendance a pour rôle d'associer à un sous-ensemble de systèmes leurs nouveaux modes de fonctionnement en fonction d'un changement de mode interne d'un système. Nous définissons une

application notée *dependency* dont le rôle est d'assurer des relations de reconfiguration entre systèmes.

**Définition 24 (Mécanisme de dépendance)**
*Soient $S$ un ensemble du système, $OM$ l'ensemble des modèles de modes de fonctionnement connexes et $T_{\rightsquigarrow}$ l'ensemble de transitions de dépendance.*

*$dependency : T_{\rightsquigarrow} \rightarrow OM^{|S|}$ est un mappage tel que $dependency(t)$ indique le mode de fonctionnement activé de chaque système de $S$ après le déclenchement de $t$.*

De nouveaux éléments sont également ajoutés à cette couche pour assurer le mécanisme de dépendance :

— la classe de couleur $ModeSoS$ telle que $ModeSoS = \cup(S_i, Mode)$ et $M$ une variable définit sur $ModeSoS$;

— la place *SoS Mode Management* définie par $ModeSoS$ et marquée par autant de jetons que de systèmes dans le SdS et chacun d'eux contient l'identité du système correspondant et son mode initial ;

— Relier les transitions de dépendance à la place *SoS Mode Management* par deux arcs. Un arc qui absorbe tous les jetons qu'elle contient et un arc qui lui génère un ensemble équivalent de jetons contenant les nouveaux modes des systèmes.

Dans l'Algorithme 7, nous créons la couche SdS qui est la quatrième couche de notre approche et qui modélise la dépendance au sein du SdS. La quatrième couche de l'exemple directeur est représentée par la Figure 4.18(a) et est obtenue après la génération de l'algorithme 7.

## 4.8 Simulation et vérification formelle

### 4.8.1 Simulation

Le modèle RdPHN de la Figure 4.1 décrit les états du SdS étudié et les événements pouvant entraîner un changement d'état. En faisant des simulations du modèle RdPHN obtenu, il est possible d'étudier différents scénarios et d'explorer les comportements du SdS. Le but de la simulation est de débloquer et de vérifier que le comportement du modèle est correct par rapport à la spécification du SdS. Par exemple, à tout moment donné un unique mode de fonctionnement est activé par système, l'état du composant commun reste cohérent après un évènnement de commutation, etc.
Pour vérifier que le modèle développé préserve le comportement souhaité pour le SdS, nous avons simulé le modèle RdPHN de manière interactive et automatique.
Il est possible d'observer les effets des différentes étapes directement sur la représentation

**Algorithme 7 :** Génération du modèle de SdS

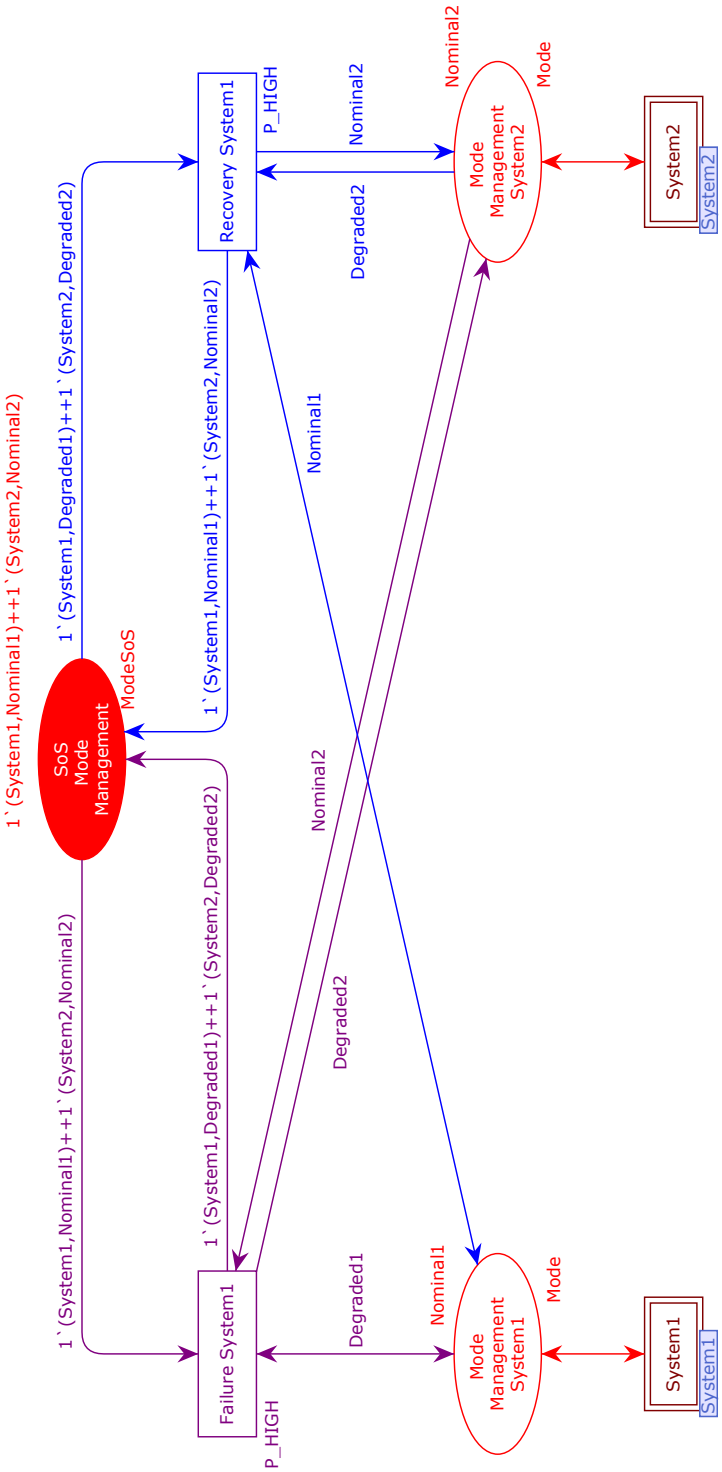
**Entrées :** l'ensemble $S$ des systèmes ; l'ensemble $M$ des modes ; l'ensemble des RdPCH $OM_{i,j}$ $(i = 1..|S|, j = 1..|M_i|)$ ;
**Output :** Modèle RdPCH final du $SdS$

Create page $SoS$;
$K = K \cup ModeSoS$;
Ajouter la place $SoS\ Mode\ Management$ ;
Ajouter les transitions $T_{\rightsquigarrow}$ ;
Ajouter les arcs $SoS\ Mode\ Management$ to $T_{\rightsquigarrow}$ ;
$M_0(SoS\ Mode\ Management) = empty$;
**pour chaque** *system,* $i \leftarrow 1$**à** $|S|$ **faire**
  $M_0(SoS\ Mode\ Management) = M_0(SoS\ Mode\ Management) + +(S_i, M_{i,1})$;
  Ajouter la transition de substitution, $T_i^{\Updownarrow}$, associée avec $S_i$ dans $SdS$;
  Ajouter la place $Mode\ Management\ S_i$ dans $SdS$;
  Ajouter l'arc connectant $Mode\ Management\ S_i$ à $T_i^{\Updownarrow}$ ;
  Ajouter l'arc connectant $Mode\ Management\ S_i$ à $T_{\rightsquigarrow}$ ;
  $W^-(Mode\ Management\ S_i, t) = M_{i,1}$;
  $W^+(Mode\ Management\ S_i, t) = target\_mode(t)$;
**fin**
**pour chaque** $t \in T_{\rightsquigarrow}$ **faire**
  $W^+(SoS\ Mode\ Management, t) = M_0(SoS\ Mode\ Management)$;
  $W^-(SoS\ Mode\ Management, t) = dependency(t)$;
**fin**

graphique du modèle RdPC.
La simulation interactive est similaire au débogage en une étape. Elle fournit un moyen de parcourir le modèle RdPHN, en examinant en détail les différents scénarios et en vérifiant si le modèle fonctionne comme prévu. Pendant une simulation interactive, le simulateur calcule l'ensemble des transitions activées dans chaque marquage rencontré. Ensuite, nous choisissons entre les transitions activées l'étape suivante. La simulation automatique est similaire à l'exécution du programme. Le but est de simuler le modèle le plus rapidement possible.

Avant et après une simulation automatique, le marquage actuel et les transitions activées sont affichés comme décrit pour le mode interactif. Cependant, le jeu de jeton ne s'affiche pas lors des simulations automatiques. Bien entendu, cela fournira généralement moins d'informations que souhaité. Une possibilité simple d'obtenir des informations sur ce qui s'est passé consiste à utiliser le rapport de simulation, qui est un fichier texte contenant des informations détaillées sur toutes les liaisons de transitions qui ont été franchies. La Figure 4.19 montre un rapport de simulation des 18 premières étapes d'une simulation automatique de notre SdS.

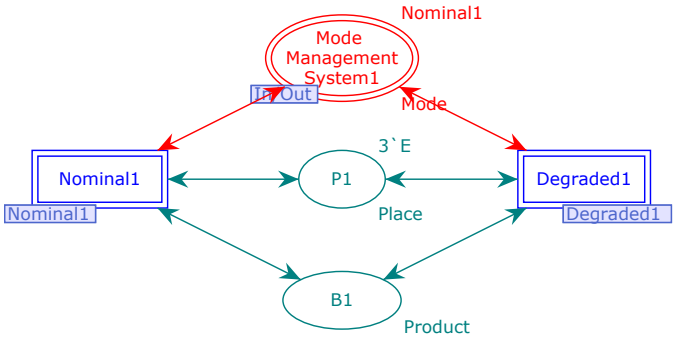
Les résultats obtenus étaient les mêmes que ceux décrit au cours de ce chapitre. Ceci



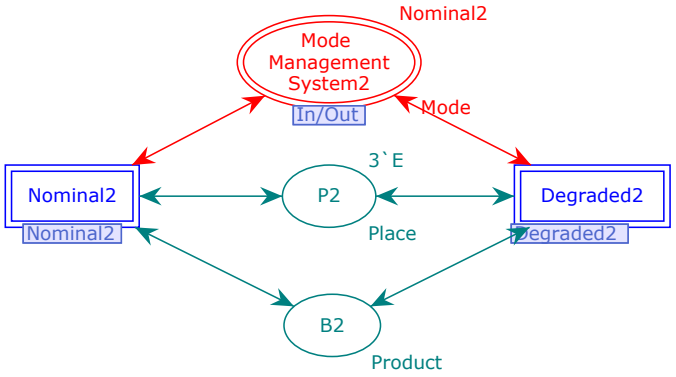

(a) Super-page SdS

nous a conforté concernant la préservation du comportement spécifié dans le cahier des charges du SdS. Ainsi à travers la simulation du modèle nous pouvons voir aussi les dépendances entre les systèmes du SdS lors le disfonctionnement d'un composant.

Au-delà de la simulation, des méthodes de vérification formelle peuvent être appliquées




(b) sous-page *System*1




(c) Sous-page *System*2

Figure 4.18 – Modèle du SdS

```
------------------------------------------------
1       0       Start_C11 @
2       0       Start_C21 @ (1:C21)
3       0       End_C11 @ (1:C11)
4       0       Start_C13 @ (1:C13)
5       0       Start_C11 @ (1:C11)
6       0       End_C21 @ (1:C21)
7       0       Start_C22 @ (1:C22)
8       0       Faillure @ (1:C11)
9       0       Failure_System1 @ (1:SoS)
10      0       Start_C23 @ (1:C23)
11      0       End_C22 @ (1:C22)
12      0       Start_C22 @ (1:C22)
13      0       End_C22 @ (1:C22)
14      0       Start_C22 @ (2:C22)
15      0       End_C13 @ (1:C13)
16      0       Recovery @ (1:C11)
17      0       Recovery_System1 @ (1:SoS)
18      0       End_C22 @ (2:C22)
```

Figure 4.19 – Example de rapport de simulation partiel

pour vérifier certaines propriétés du modèle RdPHN du SdS. La simulation ne permet pas d'étudier tous les scénarios mais le graphe d'état associé au modèle embrasse lui tout l'espace d'état accessible. Ainsi, vérifier les propriétés de sûreté sur tout le graphe est nécessaire pour s'assurer de la non occurence d'un problème spécifique. Encore faut-il être capable de caractériser l'occurence de ce problème sur un graphe d'état.

### 4.8.2 Vérification formelle

Les modèles RdPHNs sont formels, dans le sens où le langage de modélisation RdPHN a une définition mathématique pour sa syntaxe et sa sémantique. Ceci signifie qu'ils peuvent être utilisés pour vérifier les propriétés du système, c'est-à-dire prouver que certaines propriétés souhaitées sont remplies ou que certaines propriétés non souhaitées ne sont jamais vérifiées. La vérification des propriétés du système est prise en charge par un ensemble de méthodes d'espace d'état. L'idée de base des espaces d'états est de calculer tous les états accessibles et les changements d'état du modèle RdPHN et de les représenter sous la forme d'un graphe dirigé où les noeuds représentent les états et les arcs représentent les événements qui se produisent. Les espaces d'états peuvent être entièrement construits de manière automatique. À partir d'un espace d'état construit, il est possible de répondre à un grand nombre de questions de vérification concernant le comportement du système, telles que l'absence d'interblocage (ou deadlock), la possibilité de toujours pouvoir atteindre un état donné et la garantie de la fourniture d'un service donné.

À cet égard, nous avons généré le graphe d'état associé au modèle de la Figure 4.18 d'abord, et ensuite le rapport de synthèse, via l'outil CPN Tools. La Figure 4.20 représente un extrait du rapport généré.

Dans ce qui suit, nous allons nous intéresser à vérifier que le comportement du SdS respecte les règles suivantes :

**Mode activé** Pour chaque système, un seul mode de fonctionnement est activé à tout moment.

D'après le rapport, la meilleure borne entière supérieure de la place *Mode Management System1* est 1, ce qui signifie qu'il y a au plus un jeton dans la place *Mode Management System1*, et qu'il existe des marquages accessibles où il y a un jeton dans *Mode Management System1*.

D'autre part, la meilleure borne entière inférieure de la place *Mode Management System1*, et qui spécifie le nombre minimal de jetons pouvant résider sur cette place dans tout marquage accessible, est de 1 aussi, ce qui signifie qu'il y a toujours au moins un jeton dans cette place. Avec la meilleure borne supérieure de 1, ceci signifie qu'il y a exactement un jeton dans cette place pour tout marquage accessible même si la couleur de ces jetons peut varier.

```
 Statistics
------------------------------------------------------------------

  State Space                                Scc Graph
     Nodes:  5120                               Nodes:  1
     Arcs:   24704                              Arcs:   0
     Secs:   2                                  Secs:   1
     Status: Full

 Boundedness Properties
------------------------------------------------------------------

  Best Integer Bounds
                                          Upper      Lower
     C11'Init_C11 1                       1          0
     C11'Traitment_C11 1                  1          0
     C12'Init_C12 1                       1          0
     C12'Traitment_C12 1                  1          0
     C13'Init_C13 1                       1          0
     C13'Init_C13 2                       1          0
     C13'Traitment_C13 1                  1          0
     C13'Traitment_C13 2                  1          0
     C21'Init_C21 1                       1          0
     C21'Traitment_C21 1                  1          0
     C22'Init_C22 1                       1          0
     C22'Init_C22 2                       1          0
     C22'Traitment_C22 1                  1          0
     C22'Traitment_C22 2                  1          0
     C23'Init_C23 1                       1          0
     C23'Init_C23 2                       1          0
     C23'Traitment_C23 1                  1          0
     C23'Traitment_C23 2                  1          0
     SoS'Mode_Management_System1 1        1          1
     SoS'Mode_Management_System2 1        1          1
     SoS'SoS_Mode_Management 1            2          2
     System1'B1 1                         3          0
     System1'P1 1                         3          0
     System2'B2 1                         3          0
     System2'P2 1                         3          0

 Home Properties
------------------------------------------------------------------

  Home Markings
     All
```

```
Boundedness Properties
------------------------------------------------------------

  Best Upper Multi-set Bounds
     C11'Init_C11 1                    1`P11
     C11'Traitment_C11 1               1`P11
     C12'Init_C12 1                    1`P12
     C12'Traitment_C12 1               1`P12
     C13'Init_C13 1                    1`P13
     C13'Init_C13 2                    1`P13
     C13'Traitment_C13 1               1`P13
     C13'Traitment_C13 2               1`P13
     C21'Init_C21 1                    1`P21
     C21'Traitment_C21 1               1`P21
     C22'Init_C22 1                    1`P22
     C22'Init_C22 2                    1`P22
     C22'Traitment_C22 1               1`P22
     C22'Traitment_C22 2               1`P22
     C23'Init_C23 1                    1`P23
     C23'Init_C23 2                    1`P23
     C23'Traitment_C23 1               1`P23
     C23'Traitment_C23 2               1`P23
     SoS'Mode_Management_System1 1     1`Nominal1++1`Degraded1
     SoS'Mode_Management_System2 1     1`Nominal2++1`Degraded2
     SoS'SoS_Mode_Management 1         1`(System1,Nominal1)++
                                       1`(System1,Degraded1)++
                                       1`(System2,Nominal2)++
                                       1`(System2,Degraded2)
     System1'B1 1                      3`B
     System1'P1 1                      3`E
     System2'B2 1                      3`B
     System2'P2 1                      3`E

Liveness Properties
------------------------------------------------------------

  Dead Markings
     None

  Dead Transition Instances
     None

  Live Transition Instances
     All
```

C'est ce que nous devions prouver, car *Mode Management System1* est spécifié comme contenant en permanence un unique jeton. Il faut maintenant vérifier la

```
Boundedness Properties
------------------------------------------------------------------

  Best Lower Multi-set Bounds
     C11'Init_C11 1                      empty
     C11'Traitment_C11 1                 empty
     C12'Init_C12 1                      empty
     C12'Traitment_C12 1                 empty
     C13'Init_C13 1                      empty
     C13'Init_C13 2                      empty
     C13'Traitment_C13 1                 empty
     C13'Traitment_C13 2                 empty
     C21'Init_C21 1                      empty
     C21'Traitment_C21 1                 empty
     C22'Init_C22 1                      empty
     C22'Init_C22 2                      empty
     C22'Traitment_C22 1                 empty
     C22'Traitment_C22 2                 empty
     C23'Init_C23 1                      empty
     C23'Init_C23 2                      empty
     C23'Traitment_C23 1                 empty
     C23'Traitment_C23 2                 empty
     SoS'Mode_Management_System1 1       empty
     SoS'Mode_Management_System2 1       empty
     SoS'SoS_Mode_Management 1           empty
     System1'B1 1                        empty
     System1'P1 1                        empty
     System2'B2 1                        empty
     System2'P2 1                        empty
```

Figure 4.20 – Extrait du rapport d'espace d'états

couleur des jetons. Nous considérons alors le meilleur multi-ensemble supérieur des bornes de la place *Mode Management System1*. Celui-ci considère non seulement le nombre de jetons, mais également leurs couleurs. Il spécifie, pour chaque couleur de l'ensemble de couleurs de la place, le nombre maximal de jetons présents sur cette place avec la couleur donnée dans tout marquage accessible. Ceci est spécifié en tant que multi-ensemble, où le coefficient de chaque valeur est le nombre maximal de jetons avec la valeur donnée.

Ainsi, la place *Mode Management System1* a le multi-ensemble suivant comme meilleure multi-ensemble supérieure de borne :

$1`Nominal1 + +1`Degraded1$.

Ceci spécifie qu'il y a un maximum de 1 jetons avec la couleur $Nominal1$ et 1 jetons avec la couleur $Degraded1$.

À partir de ce constat, nous pouvons en conclure qu'un seul mode de fonctionnement est activé à la fois pour $System1$.

Le même résultat s'applique aussi pour $System2$ car les meilleures bornes entières supérieures et inférieures de la place *Mode Management System2* sont égales à 1 et le meilleur multi-ensemble supérieur de borne correspondant est :
$1`Nominal2 + +1`Degraded2$.

**Etat des systèmes** Au niveau de la couche SdS, l'état respectif des systèmes est toujours analytiquement calculable.

Ceci est le rôle de la place *SoS Mode Management*. Elle a une meilleure borne supérieure correspondant à la valeur 2 qui est égale à la meilleure borne inférieure, ce qui signifie que, dans tout marquage accessible, il y a exactement deux jetons dans *SoS Mode Management*. Cette valeur 2 correspond au nombre des systèmes qui compose le SdS étudié.

D'une autre côté, la place *SoS Mode Management* a le meilleure multi-ensemble supérieure de borne :
$1`(System1, Nominal1)++1`(System1, Degraded1)++1`(System2, Nominal2)++ 1`(System2, Degraded2)$.

Ceci spécifie qu'il y a un maximum de 1 jetons avec la couleur $(System1, Nominal1)$ dans *SoS Mode Management* dans tout marquage accessible (et de même pour les couleurs $(System1, Degraded1)$, $(System2, Nominal2)$, $(System2, Degraded2)$). Il spécifie également qu'il existe un marquage accessible où il y a 1 jetons avec la couleur $(System1, Nominal1)$ dans la place. Par conséquent, la place *SoS Mode Management* a toujours une visibilité sur l'état des systèmes composants en terme de mode activé.

**Réinitialisation** Chaque système du SdS doit pouvoir commuter entre ses modes de fonctionnement.

Les propriétés d'accueil nous indiquent que tous les marquages sont des *home-marking*. Un marquage a la propriété home-marking s'il peut toujours être à nouveau atteint depuis n'importe quel marquage accessible. Ceci garantie que les systèmes du SdS basculent facilement entre leurs modes de fonctionnement.

**Vivacité du modèle** Le modèle doit être vivant et sans blocage.

Le rapport indique d'une part qu'il n'y a aucun marquage dans lequel aucun élément de liaison n'est activé (pas de marquage mort) dans le modèle développé et que toutes les parties du modèle qui ne peuvent être activées (pas de transitions mortes). D'autre part, le rapport spécifie que toutes transitions sont vivantes. Une transition est vivante si, à partir d'un marquage accessible, nous pouvons toujours trouver une séquence d'occurrences contenant la transition. D'où notre modèle est bien vivant.

## 4.9 Conclusion

Ce chapitre a détaillé une démarche de modélisation hiérarchique ascendante, utilisant les RdPHNs, des SdSs dynamiques.Le démarche proposé décompose les systèmes du SdS en plusieurs modes de fonctionnement, chacun étant décomposé en composants. Ceci nous a permis de fournir une modélisation basée sur l'approche multi-modèle, une vérification et un contrôle de la dynamique interne et inter-systèmes par étapes. Dans les faits, suite à un événement exceptionnel, une défaillance par exemple, le système concerné commute de mode de fonctionnement afin de maintenir un fonctionnement acceptable. Une telle reconfiguration influence le fonctionnement global du SdS et déclenche d'autres reconfigurations dans d'autres systèmes du SdS. La première étape de la démarche est Modèle de composants où chaque composant est conçu indépendamment des autres. La deuxième étape est Modèle de modes dans laquelle les modes de fonctionnement sont étudiés et modélisés comme composition des composants de la première étape. La troisième étape porte sur la conception des systèmes avec tous leurs modes de fonctionnement ainsi que les différentes dynamiques de commutation en utilisant la TCS. Enfin, la dernière étape consiste à modéliser les dépendances entre les systèmes. En effet, si un système change de mode, d'autres systèmes doivent également changer de modes pour s'adapter aux changements.
Nous avons illustré notre démarche par un SdS comportant deux systèmes. Le modèle RdPHN obtenu a été exploité pour valider une série d'exigences sources en s'appuyant sur les services d'analyse fournis par l'outil de modélisation utilisé.

Ce faisant, nous avons donc proposé une démarche outillée de construction sûre des SdS dynamiques. Dans les chapitres qui suivent, nous étudierons deux applications différentes de systèmes critiques pour montrer la plus-value effective et la capacité à traiter des problèmes assez variés de nos travaux.

# Chapitre 5

# Franchissement de la frontière sous ERTMS

## 5.1 Introduction

L'interopérabilité des systèmes ferroviaires est très importante en Europe car elle constitue un enjeu économique majeur pour le trafic transeuropéen. Cependant, le trafic ferroviaire ininterrompu et continu sur l'ensemble du réseau ferroviaire européen exige des caractéristiques de conformité de l'infrastructure et des véhicules. Ainsi, franchir une frontière sans reconfiguration lourde constitue un objectif important dans le domaine ferroviaire. Comme indiqué précédemment, le concept de SdS et la gestion des modes de fonctionnement sont utilisés dans ce travail pour assurer l'harmonisation de la capacité technique au sein du système ferroviaire européen. Dans ce chapitre, nous présentons une approche de sécurité outillée en vue de franchir les frontières européennes sous ERTMS Niveau 2.

## 5.2 Contexte du franchissement de la frontière

L'interopérabilité vise à créer un réseau ferroviaire permettant un transport sûr, conforme au niveau de performance requis des lignes et ne nécessitant pas de transfert de train. Ceci nécessite le respect d'un ensemble de règles, de conditions techniques et opérationnelles garantissant le respect des exigences essentielles. La proposition européenne correspondante repose sur un système européen de signalisation ferroviaire appelé ERTMS. Néanmoins, la gestion concrète des informations de signalisation ferroviaire utilisées par ERTMS doit respecter des règles propres à chaque pays. Le réseau ferroviaire européen repose sur l'interconnexion physique et logique des réseaux ferroviaires nationaux où seulement une partie concernant le matériel roulant et ses échanges avec le sol sont normalisés dans ERTMS.

Clairement, le comportement résultant du système sol est spécifique à chaque pays et il n'est pas inclus dans la spécification ERTMS. Ainsi, au niveau d'un passage frontalier, le système sol change et certains modes opératoires ERTMS (Voir sous-section 2.2.3) interdits dans un pays sont autorisés dans un autre. Les vitesses maximales autorisées dans un pays donné peuvent également être différentes. Le fait de franchir une frontière peut être considéré comme instantané. En suivant cette considération, [Kadri et al. 2014] ont proposé que le train puisse franchir la frontière selon un mode opératoire commun aux configurations spécifiques des deux pays. Cette condition n'est pas suffisante car, dans le même mode opératoire, il existe également des contraintes spécifiques à chaque pays et le train doit toutes les remplir lors du passage de la frontière. Ceci pose un problème de contrôle qui spécifie le mode transitoire conduisant à un état satisfaisant les contraintes du pays actuel et se terminant à un point où les contraintes des pays d'origine et d'arrivée sont respectées. En effet, on ne peut considérer le passage d'une frontière par un train d'une centaine de mètres comme instantané.
Trouver un moyen sûr pour franchir les frontières européennes en utilisant ERTMS devient donc un sujet d'intérêt. Ainsi, l'objectif de ce chapitre est de proposer une approche sûre pour contrôler le franchissement des frontières par des trains à travers l'UE en utilisant ERTMS Niveau 2.

## 5.3 Modèles ERTMS

Plusieurs modèles de la plateforme ferroviaire ERTMS ont été proposés dans la littérature. [Hermanns et al. 2005] ont utilisé des StoCharts pour modéliser l'ETCS et évaluer la sûreté de fonctionnement du système radio de train. Les StoCharts sont l'extension orientée qualité de service QoS (Quality of Service) des Statecharts UML (Unified Modeling Language). Puisqu'ils manquent d'outils, ils sont traduits dans le langage de modélisation et de description pour systèmes stochastiques et temporisés MoDeST (Modeling and Description Language for Stochastic and Timed Systems), qui est un langage formel utilisé pour décrire les systèmes temporisés stochastiques. [Vernez et Vuille 2009] ont proposé une approche fonctionnelle d'analyse du mode de défaillance afin d'optimiser la sûreté de fonctionnement du système ERTMS Niveau 2. Le modèle de base prend en compte les procédures opérationnelles ainsi que le fonctionnement en mode dégradé. Il est mis en oeuvre sur un outil logiciel commercial. Cette approche évalue le niveau de risque global et le niveau de disponibilité du système de signalisation ferroviaire complexe et identifie ses vulnérabilités. [Lalouette et al. 2010] ont proposé une approche basée sur les RdPCs pour évaluer la sûreté de fonctionnement du système ERTMS superposé au système français. Cette approche évalue un ensemble d'aléas susceptibles d'être rencontrés au cours du cycle de vie opérationnel d'un système et non plus de profils de missions choisis arbi-

trairement parmi un ensemble de trajectoires possibles du système. [Qiu et al. 2014] ont considéré le système ERTMS Niveau 2 comme un SdS et ont proposé une méthodologie de conception à l'aide de Statecharts et des réseaux valués VBS (Valuation-Based System). Cette méthodologie propose un modèle dysfonctionnel du SdS global intégrant les aspects matériels et réseaux et le facteur humain. [Herranz et al. 2011] ont modélisé le système ERTMS/ETCS à l'aide de diagrammes UML, puis ils ont transformé leurs modèles UML en spécifications Uppaal (environnement intégré pour la modélisation, la validation et la vérification de systèmes temps réel). [Bougacha et al. 2019] ont proposé une approche d'ingénierie dirigée par les modèles pour la modélisation et la vérification des systèmes de signalisation ferroviaire, qui envisage d'abord de modéliser graphiquement le système avec un diagramme de classe UML étendu par des aspects de sécurité, puis de générer un modèle Event-B avec des transformations de modèles à modèles (M2M). L' [Agence ferroviaire européenne] a financé un projet de formalisation et de validation de chemins de fer européens qui proposait l'utilisation de Rational tools pour la formalisation et la validation des spécifications de l'ETCS. [Zimmermann et Hommel 2003] ont utilisé une modélisation par réseaux de Petri stochastiques pour modéliser et évaluer le comportement en cas de défaillance et de récupération de la liaison de communication, ainsi que sa combinaison avec l'échange d'informations vitales sur les trains entre des trains et des centres de blocs radio.
Dans ce chapitre, nous considérons le système ERTMS Niveau 2 comme un SdS et nous utilisons les RdPCs pour modéliser le comportement dynamique des trains lors du franchissement de la frontière. Il est à noter qu'aucun des travaux cités n'a analysé le comportement des trains lors de changements du système sol sachant qu'actuellement les trains ne franchissent généralement pas la frontière en utilisant le système ERTMS.

## 5.4 Modélisation du système ERTMS

### 5.4.1 Décomposition modale

Le système ERTMS Niveau 2 est un système critique, réparti et complexe. Il est considéré comme un SdS dans la mesure où il est composé de deux systèmes complexes, indépendants et hétérogènes : le système sol et le système bord (voir section 2) qui assurent la surveillance continue des mouvements des trains avec une communication continue.
Le système bord est largement intégré dans les trains qui traversent l'UE ainsi chaque pays de l'UE gère le système sol installé sur son infrastructure ferroviaire indépendamment des autres pays. Dans ce qui suit, nous confondons le système ferroviaire de l'UE équipée du système ERTMS et le système ERTMS. Ceci est pour simplifier le propos.

**Définition 25 (Ensemble des trains)**
*L'ensemble des trains est appelé* $Tr = \{Tr_1, Tr_2, ...Tr_{|Tr|}\}$ *où* $|Tr| > 1$.

**Définition 26 (Ensemble des systèmes sols)**
*L'ensemble des systèmes sols est appelé* $Ts = \{Ts_1, Ts_2, ...Ts_{|Ts|}\}$ *où* $|Ts| > 1$.

**Définition 27 (Système ERTMS)**
*L'ensemble des systèmes du système ERTMS est appelé* $SoS = Tr \cup Ts$.

En se focalisant sur la maîtrise de comportement des trains aux frontières, le système ERTMS est étudié en tant que SED afin de se servir d'une des notions couramment utilisée pour la conception de la commande des SED, la notion de mode de fonctionnement. Elle permet de diminuer considérablement la complexité du système par la décomposition.
Appliqué à notre SdS, un mode de fonctionnement est un processus de fonctionnement entre un train et un système sol en appliquant de manière conventionnelle la TCS. À notre connaissance, la notion de mode de fonctionnement n'a jusqu'à présent pas été utilisé pour les SdSs.
Le système sol a pour rôle la gestion des autorités de mouvement, qui les envoie aux trains, ainsi que les informations nécessaires au calcul du profil de vitesse approprié. Pour garantir l'interopérabilité lors de franchissement des frontières, notre proposition consiste aussi à définir des modes pour chaque système sol : un mode nominal, représentant les règles d'exploitation nationaux, des modes transitoires dans les zones frontalière dans lesquels le système sol assure non seulement ses contraintes de circulation mais aussi celles du système sol voisin.

**Définition 28 (Modes de système sol)**
*Soit* $Ts_j$ *un système sol où* $j = 1..|Ts|$.
*L'ensemble des modes du système sol* $Ts_j$ *est noté* $\{M_{j,1}, M_{j,2}, ...M_{j,|Ts_j|}\}$ *tel que* $|Ts_j| \geq 1$.

**Définition 29 (Modes de fonctionnement)**
*L'ensemble des modes de fonctionnement du système ERTMS est appelé* $OM$ *où*

$$\begin{array}{ccc} Tr \times Ts & \longrightarrow & OM \\ (Tr_i, M_{j,k}) & \longmapsto & OM_{i,(j,k)} \end{array}$$

*est une application tel que* $OM_{i,(j,k)}$ *indique le mode de fonctionnement associé à un train* $Tr_i$ *fonctionnant en mode* $M_{(j,k)}$ *du système sol* $Ts_j$ .

De point de vue modélisation, nous adoptons l'approche multi-modèle. Elle nous a permis d'associer pour chaque mode de fonctionnement $OM_{i,(j,k)}$ tel que $i \in 1..|Tr|$, $j \in 1..|Ts|$ et $k \in 1..|Ts_j|$ un modèle RdPC

$< P_{i,(j,k)}, T_{(j,k)}, k_{i,(j,k)}, D_{i,(j,k)}, W^-_{i,(j,k)}, W^+_{i,(j,k)}, \phi_{i,(j,k)}, M_{i,(j,k)_0} >$.
Pour des raisons de simplicité, nous donnons la même identité du mode de fonctionnement $OM_{i,(j,k)}$ au modèle RdPC associé.

**Remarque 3** *Comme les trains représentent les systèmes dynamiques au sein de SdS, ils seront modélisés dans les modèles $OM_{i,(j,k)}$ par des jetons.*

D'un autre côté, tous les trains qui circulent simultanément dans un même mode sol, sont soumis aux mêmes exigences, normes et règles nationales. Par conséquent, des modes de fonctionnement similaires s'observent. Dans le but de profiter de cette similitude pour simplifier la modélisation des modes de fonctionnement, nous définissons le concept de mode de fonctionnement abstrait représentant le comportement des trains du même mode d'un système sol donné. Ce concept permet aussi de réduire la taille et la complexité du modèle ERTMS final.

**Définition 30 (Mode de fonctionnement abstrait)**
*$OM_{j,k} =< P_{j,k}, T_{j,k}, k_{j,k}, D_{j,k}, W^-_{j,k}, W^+_{j,k}, \phi_{j,k}, M_{j,k_0} >$, $j \in 1..|Ts|$ et $k \in 1..|Ts_j|$ est appelé mode de fonctionnement abstrait si, et seulement si, pour tout mode de fonctionnement $OM_{i,(j,k)} =< P_{i,(j,k)}, T_{(j,k)}, k_{i,(j,k)}, D_{i,(j,k)}, W^-_{i,(j,k)}, W^+_{i,(j,k)}, \phi_{i,(j,k)}, M_{i,(j,k)_0} >$,$i \in 1..|Tr|$,*

$$< P_{j,k}, T_{j,k}, k_{j,k}, D_{j,k}, W^-_{j,k}, W^+_{j,k}, \phi_{j,k}, M_{j,k_0} >=$$

$$< P_{i,(j,k)}, T_{(j,k)}, k_{i,(j,k)}, D_{i,(j,k)}, W^-_{i,(j,k)}, W^+_{i,(j,k)}, \phi_{i,(j,k)}, M_{i,(j,k)_0} >$$

$$et\ M_{j,k_0} \subseteq M_{i,(j,k)_0}.$$

Afin de faciliter la construction des modèles des modes de fonctionnement abstraits, l'Algorithme 8 énonce les étapes à suivre.

### 5.4.2 Commutation de mode

La commutation de mode se déclenche, pour un train donné, dans deux situations : lorsqu'il s'apprête à traverser la frontière ou à la frontière. Techniquement, c'est le passage du train sur une zone spécifique de sol qui déclenche la commutation de mode pour ce train. Ceci provoque la sortie de son mode de fonctionnement actuel et l'entrée d'un second.
Dans notre approche, la commutation de mode est obtenue par le biais des transitions RdPC spécifiques dans chaque modèle de mode.

**Définition 31 (Transition de commutation de mode)**
*Soient $OM_{i,(j,k)}$ et $OM_{i,(l,m)}$ deux modes de fonctionnement tel que $k \neq m$ si $j = l$ et soient $T_{i,(j,k)}$ et $T_{i,(l,m)}$ les ensembles de transitions connexes.*

**Algorithme 8 :** Génération des modèles des modes de fonctionnement abstraits.

**Entrées :** l'ensemble $Tr$ des trains ;
l'ensemble $Ts$ des sols ;
l'ensemble $OM_{i,(j,k)}$ des mode de fonctionnement ;
**Output :** les modèles RdPCs des modes de fonctionnement abstraits
$OM_{j,k} =< P_{j,k}, T_{j,k}, k_{j,k}, D_{j,k}, W^-_{j,k}, W^+_{j,k}, \phi_{j,k}, M_{j,k_0} >$

```
pour chaque systeme Ts_j, j ← 1à |Ts| faire
    pour chaque mode M_{j,k}, k ← 1à |Ts_j| faire
        Construire le modèle OM_{j,k};
        pour chaque train Tr_i, i ← 1à |Tr| faire
            soit p la place censé contenir les trains;
            si Tr_i ∈ OM_{i,(j,k)} alors
                M_{j,k_0}(p) = M_{j,k_0}(p)∪ le jeton représentant Tr_i;
            fin
        fin
    fin
fin
```

*Si* $\exists t = T_{i,(j,k)} \cap T_{i,(l,m)} \neq \varnothing$ *alors* $t$ *correspond à une transition de commutation entre* $OM_{i,(j,k)}$ *et* $OM_{i,(l,m)}$.

### 5.4.3 Construction du modèle ERTMS

Sur la base des concepts précédents, nous sommes en mesure de définir, via un modèle unique, la gestion du trafic ferroviaire sous ERTMS.
Un système ERTMS est un RdPC $< P, T, K, D, W^-, W^+, \phi, M_0 >$ où

$P = \cup_{(j=1..|Ts|,k=1..|Ts_j|)} P_{j,k}$;

$T = \cup_{(j=1..|Ts|,k=1..|Ts_j|)} T_{j,k}$;

$K = \cup_{(j=1..|Ts|,K=1..|Ts_j|)} K_{j,k}$;

$D$ est défini à partir de $P \cup T$ dans un ensemble fini ;

$W^-, W^+$

— $\forall OM_{j,k} \in OM, \forall (p,t) \in P_{j,k} \times T_{j,k}$,

$$W^-(p,t) = W^-_{j,k}(p,t) \text{ et } W^+(p,t) = W^+_{j,k}(p,t);$$

— $\forall (OM_{j,k}, OM_{m,n}) \in OM \times OM((j \neq m) \text{ ou } (k \neq n))$,
$\forall (p,t) \in P_{j,k} \times T_{m,n}$ :
si $(p \notin P_{j,k}$ ou $t \notin T_{m,n})$ alors $W^-(p,t) = W^+(p,t) = 0$;

$\phi$ $\forall OM_{j,k} \in OM, \forall t \in T_{j,k}, \phi(t) = \phi_{j,k}(t)$ ;

$M_0$ $\forall OM_{j,k}$ $M_0(p) = M_{j,k_0}(p)$.

Pour faciliter la construction du modèle ERTMS global, nous développons un l'Algorithme 9 générant un seul RdPC à partir d'un ensemble de ses modèles de mode de fonctionnement.

**Algorithme 9 :** Génération du modèle ERTMS.

**Entrées :** l'ensemble $Tr$ des trains ;
l'ensemble $Ts$ des sols ;
l'ensemble $OM_{j,k}, (j = 1..|Ts|, k = 1..|Ts_j|)$ des modèles RdPCs des modes de fonctionnement abstraits ;
**Output :** Le modèle RdPC du système ERTMS $=< P, T, K, D, W^-, W^+, \phi, M_0 >$

$P \leftarrow \varnothing$;
$T \leftarrow \varnothing$;
$K \leftarrow \varnothing$;
$M_0 \leftarrow \varnothing$;

**pour chaque** *mode de fonctionnement abstrait* $OM_{j,k}, j \leftarrow 1$**à** $|Ts|, k \leftarrow 1$**à** $|Ts_j|$ **faire**
- $P \leftarrow P \cup P_{j,k}$;
- $T \leftarrow T \cup T_{j,k}$;
- $K \leftarrow K \cup K_{j,k}$;
- $W^-(p,t) \leftarrow W^-_{j,k}(p,t)$;
- $W^+(p,t) \leftarrow W^+_{j,k}(p,t)$;
- **pour chaque** $t \in T_{j,k}$ **faire**
  - $\phi(t) \leftarrow \phi_{j,k}(t)$ ;
- **fin**
- **pour chaque** $p \in P_{j,k}$ **faire**
  - $M_0(p) \leftarrow M_0(p) \cup M_{j,k_0}(p)$ ;
- **fin**
- **pour chaque** *mode de fonctionnement abstrait* $OM_{m,n}$, $m \leftarrow 1$**à** $(m-1), n \leftarrow 1$**à** $(k-1)$ **faire**
  - **pour chaque** $(p,t) \in P_{j,k} \times T_{m,n}$ **faire**
    - **si** $p \notin P_{j,k} \vee t \notin T_{m,n}$ **alors**
      - $W^-(p,t) \leftarrow 0$;
      - $W^+(p,t) \leftarrow 0$ ;
    - **fin**
  - **fin**
- **fin**

**fin**

**Remarque 4**

*Nous pouvons noter que l'Algorithme 9 se termine correctement car nous traitons avec un ensemble fini de modes de fonctionnement abstrait.*

**Remarque 5**

*Cet algorithme peut être traduit en machines B abstraites à l'aide de [Bon et Collart-Dutilleul 2013], [Sun et al. 2015] et [Boudi et al. 2017]. Les machines B abstraites constitueront le point de départ d'une ingénierie logicielle système critique, basée sur des méthodes formelles, comme le recommande la norme 50128 CENELEC.*

## 5.5 Application

Comme indiqué dans la section 5.2, même en utilisant le même système bord, le contexte de fonctionnement des trains peut changer en raison de règles spécifiques applicables dans un pays donné. Par exemple, les lois relatives aux vibrations et au bruit peuvent interdire l'utilisation d'une vitesse techniquement possible. Comme cela est techniquement possible, il peut être utilisé en toute sécurité dans un pays voisin. Dans ce qui suit, un exemple illustrant les deux aspects d'un ensemble différent de traitements en mode d'opération légal et de différentes vitesses maximales autorisées est présenté.

### 5.5.1 Description du système

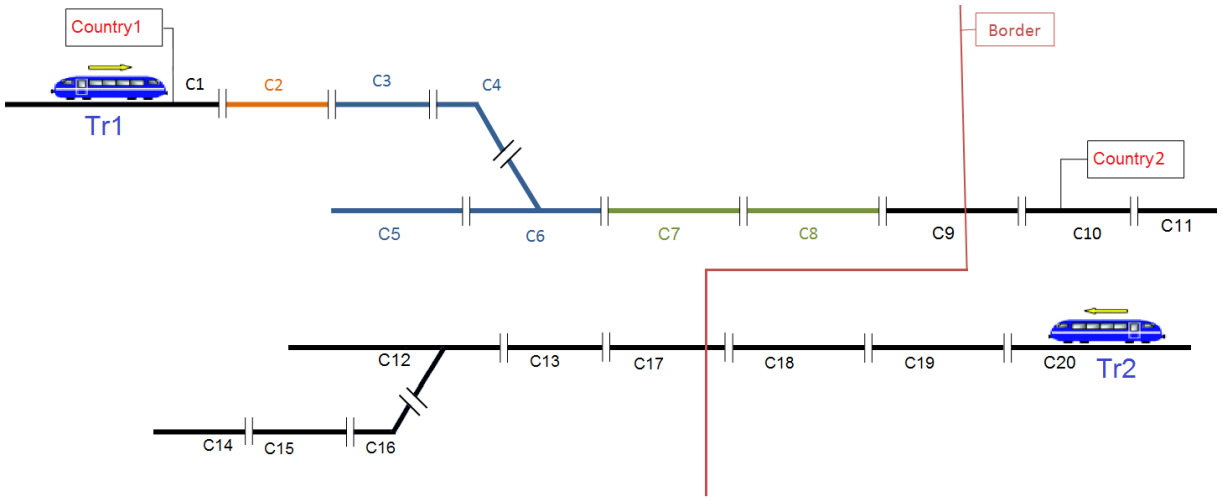



Figure 5.1 – Exemple de chemin de fer

Nous illustrons notre contribution avec un exemple simple du système ERTMS constitué de deux trains $Tr_1$ et $Tr_2$, deux systèmes sols $Ts_1$ et $Ts_2$ associés respectivement à $Country1$ et $Country2$. $Ts_1$ et $Ts_2$ ont chacun deux modes de sol : un mode national (nominal) et un mode transitoire permettant de traverser la frontière entre $Country1$ et $Country2$. Pour $Ts_1$, les modes sol sont $Cn_1$ et $TmCn_1toCn_2$ alors ceux de $Ts_2$ sont $Cn_2$ et $TmCn_2toCn_1$.

La Figure 5.1 représente une partie d'un réseau ferroviaire indiquant la position de $Tr_1$ et

de $Tr_2$. Les modes opératoires autorisés dans $Country1$ sont $[FS, RV, NL, SR, OS, SF]$ et de l'autre côté sont $[FS, SR, OS, SH]$.
Dans $Country1$, les trains circulent en mode nominal $Cn_1$ mais avant de franchir la frontière de $Country2$, les trains basculent vers le mode transitoire $TmCn_1toCn_2$ afin de satisfaire aussi les contraintes du pays destination. De même pour $Country2$, les trains circulent en mode nominal $Cn_2$ et le mode transitoire est $TmCn_2toCn_1$.
Le franchissement de la frontière est souvent à grande vitesse (en mode $FS$) mais la limite de vitesse varie considérablement d'un pays à l'autre. Dans notre exemple, la vitesse maximale autorisée dans $Country1$ est de 320 $km/h$, pendant que dans $Country2$, elle est de 250 $km/h$. Afin de garantir le respect des limitations de vitesse, les trains du côté $Country1$ devraient ralentir avant de franchir la frontière de $Country2$ si leurs vitesses dépassent les 250 $km/h$ tout en suivant une courbe de freinage (voir la Figure 5.2). La courbe de freinage a la forme issu du tableau de vitesse/distance du train qui ralentit. En effet, $Tr_1$ est soumis à une décélération avant le passage de la frontière : atteignant le segment du sol $C2$, le train se trouve dans la zone d'attention (la zone orange des Figures 5.1 et 5.2) dans laquelle les trains se déplacent vers un contrôle de vitesse et une distance unifiés. Ensuite, et pendant les 3 segments de sol suivants (les zones bleues), la vitesse de $Tr_1$ diminue par rapport à la distance en maintenant la vitesse du train dans les limites appropriées. Enfin, les segments de voie $C7$ et $C8$ (zones vertes) constituent une zone de stabilisation de la vitesse des trains avant le passage de la frontière.
Notons que l'utilisation de la vitesse maximale est spécifique à chaque pays. Par exemple, la spécification ERTMS exige une vitesse maximale de 50 $km/h$ en mode $SR$ ou $OS$, mais dans certains pays, elle atteint les 30 $km/h$. Dans ce dernier cas, l'utilisation réelle de la mise en oeuvre du système ERTMS au niveau national s'applique à 30 $km/h$, car la limitation de vitesse correspond à une loi nationale.

### 5.5.2 Modèles RdPC

D'après les définitions 25, 26 et 27, nous pouvons établir l'ensemble des trains $Tr = \{Tr_1, Tr_2\}$, l'ensemble des systèmes sols $Ts = \{Ts_1, Ts_2\}$ et l'ensemble des systèmes ERTMS $SoS = Tr \cup Ts = \{Tr_1, Tr_2, Ts_1, Ts_2\}$. L'ensemble des modes de système sol $Ts_1$ est $M_1 = \{Cn_1, TmCn_1toCn_2\}$ et celui de $Ts_2$ est $M_2 = \{Cn_2, TmCn_2toCn_1\}$.
L'ensemble des modes de fonctionnement est défini par la définition 29. Nous avons $OM = \{(Tr_1, Cn_1), (Tr_1, TmCn_1toCn_2), (Tr_1, Cn_2), (Tr_2, Cn_2), (Tr_2, Cn_1), (Tr_2, TmCn_2toCn_1)\}$.
Nous pouvons déduire, par le biais de la définition 30, que le système possède quatre modes de fonctionnement abstraits qu'on appelle $Nominal1$, $Transient1$, $Nominal2$ et $Transient2$ correspondant respectivement aux modes des systèmes sol $Cn_1$, $TmCn_1toCn_2$, $Cn_2$, et $TmCn_2toCn_1$.

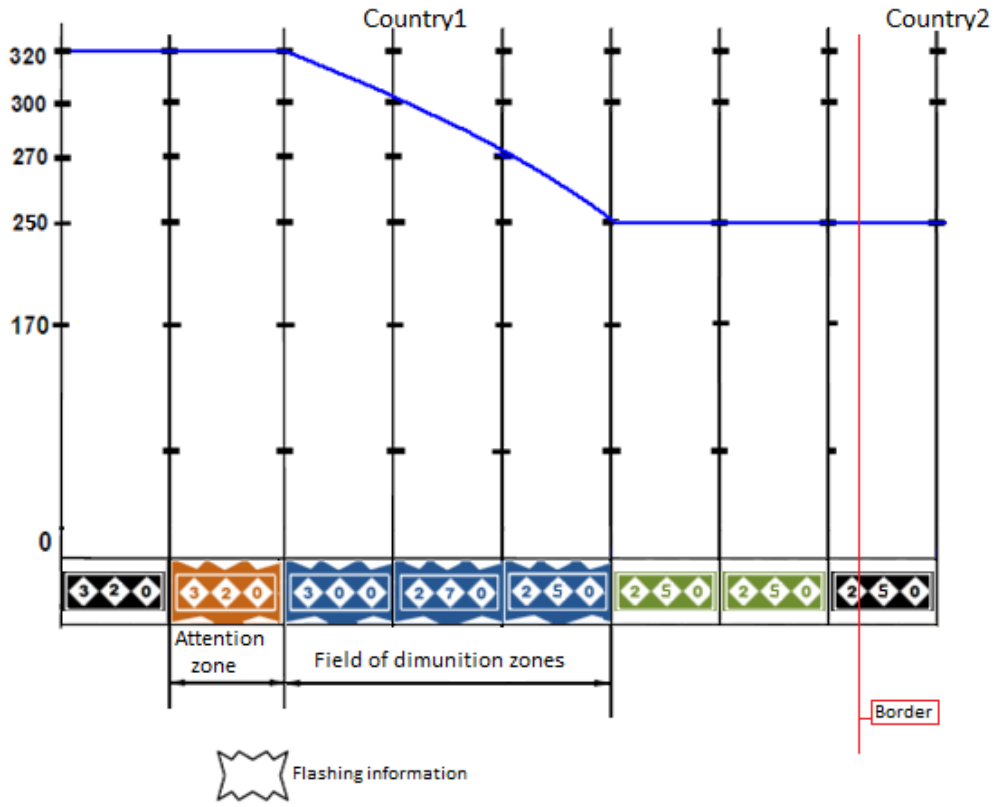



Figure 5.2 – Exemple de courbe de freinage du train

Pour mieux comprendre le modèle global, nous allons décrire d'abord les modèles RdPCs correspondants aux modes de fonctionnement abstraits.

### Modèles RdPC des modes nominaux

Les modèle des Figures 5.3 et 5.4 représentent le mode nominal de *Country*1 et *Country*2 respectivement. Dans la Figure 5.3 (resp. 5.4), le comportement interne est composé de deux places *Trains _ Cn1* et *Mode and Limited Speed of Cn1* (resp. *Trains _ Cn2* et *Mode and Limited Speed of Cn2*) et d'une transition *Movement _ Cn1* (resp. *Movement _ Cn1*) permettant aux trains d'avancer et de changer de modes. Les trains en marche sont modélisés par les jetons de la place *Trains _ Cn1* (resp. *Trains _ Cn2*). Chaque jeton a une structure composée de cinq éléments : l'identité du train, l'itinéraire, la vitesse maximale, le mode actuel et le pays actuel. Le mouvement du train est assuré par la transition *Movement _ Cn1* (resp. *Movement _ Cn2*) avec la possibilité de passer d'un mode de fonctionnement à un autre autorisé via la place *Mode and Limited Speed of Cn1* (resp. *Mode and Limited Speed of Cn2*).
La commutation de mode est représentée par les deux transitions *Crossing the Country1 Border* et *Transient mode to Cross the Country 2 Border* (resp. *Crossing the Country2 Border* et de *Transient mode to Cross the Country 1 Border*). Le premier permet aux

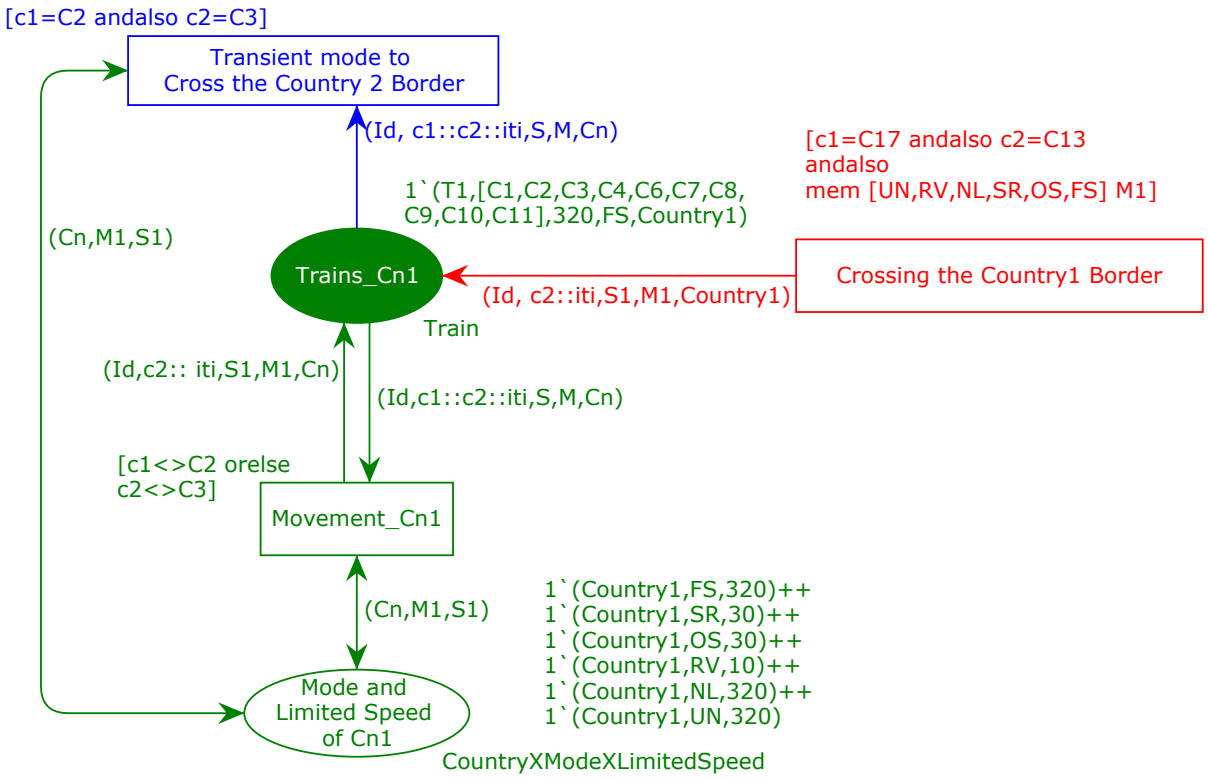



Figure 5.3 – Modèle du mode $Nominal1$

trains d'entrer dans mode actuel et le second d'en sortir.
Notons qu'à l'aide de la notion de mode de fonctionnement abstrait, nous pouvons déterminer des modes génériques représentant tous le fonctionnement des trains dans un mode donné. Par exemple, privé des jetons de la place *Trains _ Cn1*, le modèle $Nominal1$ représente un comportement générique des trains circulant du côté Country1 en mode national.

### Modèles RdPC des modes transitoires

Les Figures 5.5 et 5.6 représentent les deux modes transitoires qui démarrent dans des zones spécifiques avant de franchir la frontière. Ces deux modes permettent aux trains de circuler dans le respect des contraintes des pays des deux côtés de la frontière. Le comportement interne des modes transitoires est similaire au modèle des modes nationaux, à l'exception des jetons des places *Mode and Limited Speed TMCn2* et *Limited Speed TMCn1* qui sont déterminés à partir de l'intersection des jetons des places de *Mode and Limited Speed of Cn1* et *Mode and Limited Speed of Cn2*.
En plus, le modèle de la Figure 5.6 contient un mécanisme de décélération en mode $FS$ afin d'adapter la vitesse des trains aux contraintes du pays de destination tout en respectant la courbe de freinage de la Figure 5.2. Il est composé de la transition *Speed Diminution* et

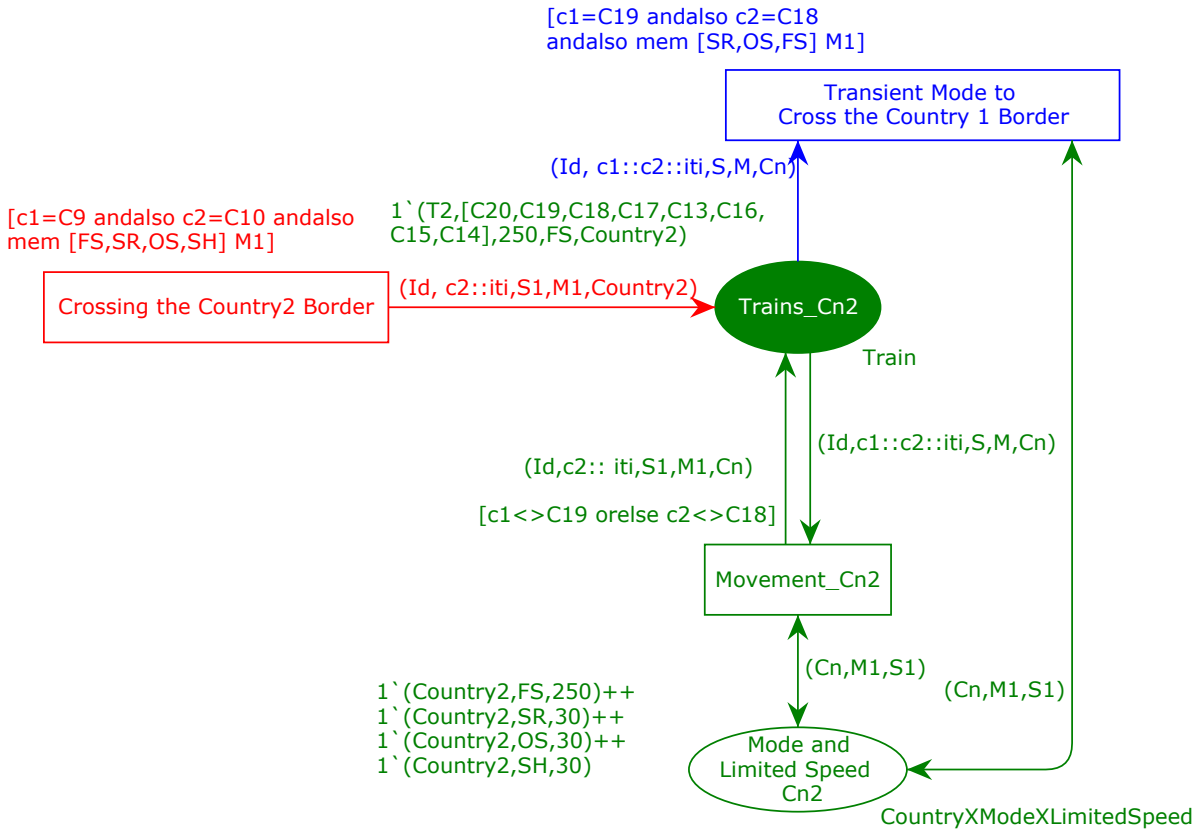



Figure 5.4 – Modèle du mode *Nominal*2

de la place *Limited Speed of Contans*, les trains doivent respecter la vitesse maximale de la zone fournie par la place *Limited Speed of Contans*.
L'accès et la sortie de mode transitoire du *Country*1 (resp. *Country*2) est assuré par les deux transitions *Crossing the Country1 Border* et *Transient mode to Cross the Country 2 Border* (resp. *Crossing the Country2 Border* et *Transient mode to Cross the Country 2 Border*).
De manière analogue aux modes nominals, nous pouvons remarquer que les modes transitoires représentent des comportements génériques des trains en mode transitoire si nous ne prenons pas en compte le jeton de la place *Trains _ TMCn1* (resp. *Trains _ TMCn2*).

#### Modèle RdPC global

À partir d'une telle spécification, nous pouvons alors obtenir le RdPC global décrivant la gestion du mode de fonctionnement de notre SdS étudié.

### 5.5.3 Simulation et verification formelle

Le modèle représentant le comportement global de notre SdS étudié a été développé en utilisant l'outil CPN Tools. Cet environnement offre plusieurs possibilités d'exploitation

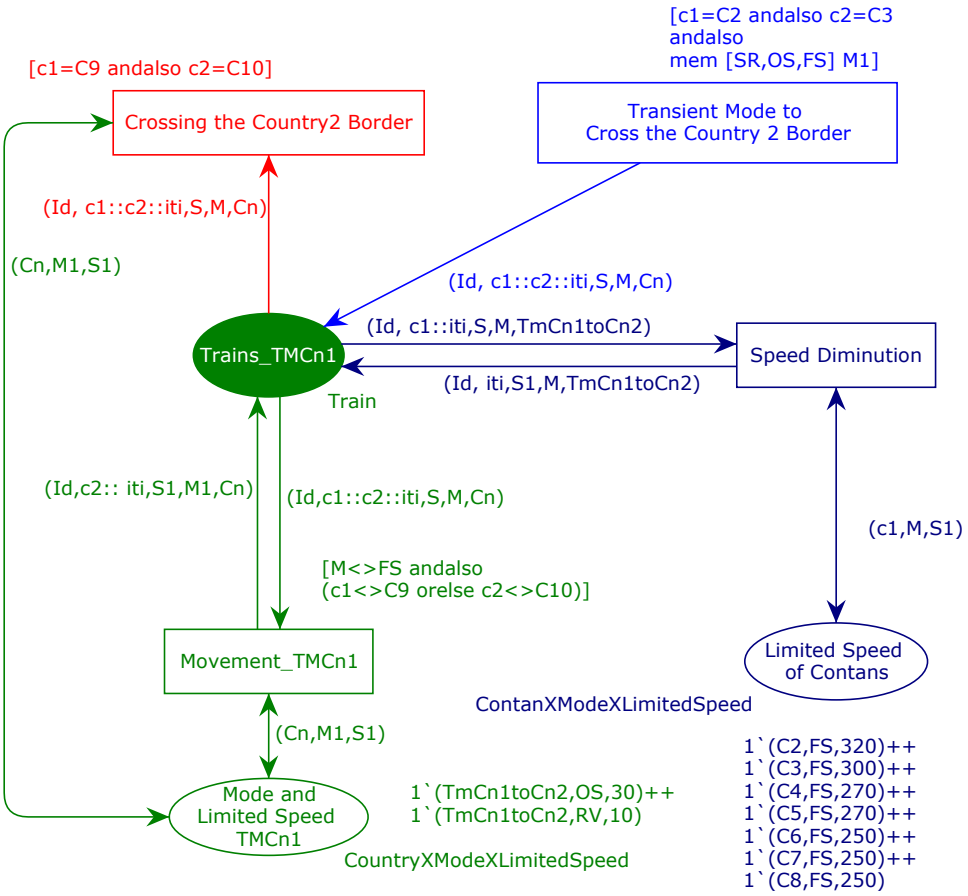



Figure 5.5 – Modèle du mode *Transient*1

des modèles conçus. Il est possible d'étudier le comportement du système modélisé à l'aide de la simulation et de vérifier les propriétés à l'aide de méthodes d'espace d'états et de vérification de modèle.

### Simulation

La simulation du modèle de la Figure 5.7 a permis d'explorer les comportements du système et de vérifier si le comportement du modèle est correct par rapport à la spécification. Par exemple, un train ne peut être que dans un seul mode de fonctionnement à un moment donné, la commutation de mode se déclenche au bon emplacement du train, désactive son mode de fonctionnement actuel et active le nouveau mode de fonctionnement approprié, etc. De plus, CPN Tools offre la possibilité de définir des moniteurs qui observent la simulation du modèle. Ces moniteurs peuvent être très utiles pour collecter des chiffres sur le transport ferroviaire entre *Country*1 et *Country*2, et plus généralement sur une échelle européenne. Par exemple, il suffit d'initialiser un moniteur comptant le nombre de tirs des transitions qui modélisent le franchissement d'un train la frontière.

Au-delà de la simulation, le modèle RdPC développé offre la possibilité d'appliquer

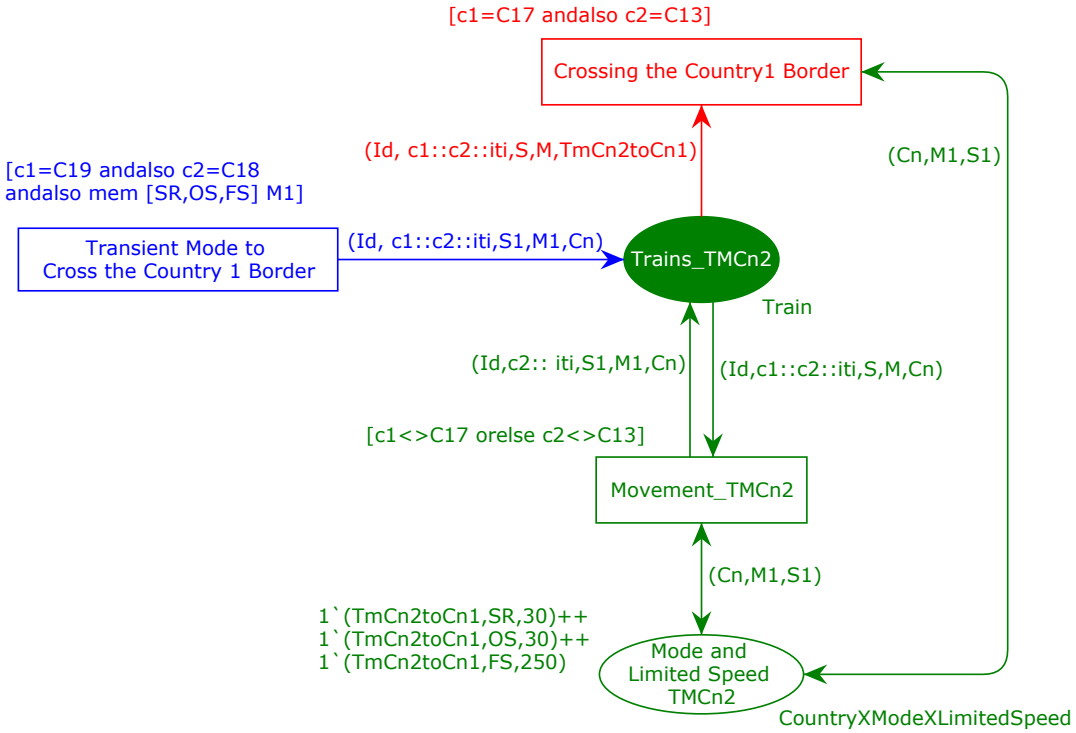



Figure 5.6 – Modèle du mode $Transient2$

des méthodes de vérification formelle pour vérifier certaines propriétés du modèle. La simulation ne permet d'observer que des scénarios particuliers d'exécution. En étudiant le graphe d'états associé au modèle global, on peut observer toutes ses séquences d'exécution. Ainsi, en vérifiant des propriétés de sûreté sur tout le graphe, le concepteur sera certain qu'aucun problème logique ne surviendra lors du fonctionnement, même après déploiement.

### Vérification formelle

De nombreuses techniques de vérification sont basées sur l'espace d'état. En utilisant l'espace d'état d'un modèle, il est possible d'examiner d'importantes propriétés comportementales, telles que la vivacité, l'équité, la sécurité, ainsi que de rechercher des défauts (par exemple, impasses, divergences, violations de l'exclusion mutuelle, etc.).
L'espace d'état du modèle de la Figure 5.7 est calculé automatiquement via CPN Tools. Un rapport de synthèse est généré par la suite et représenté par la Figure 5.8.

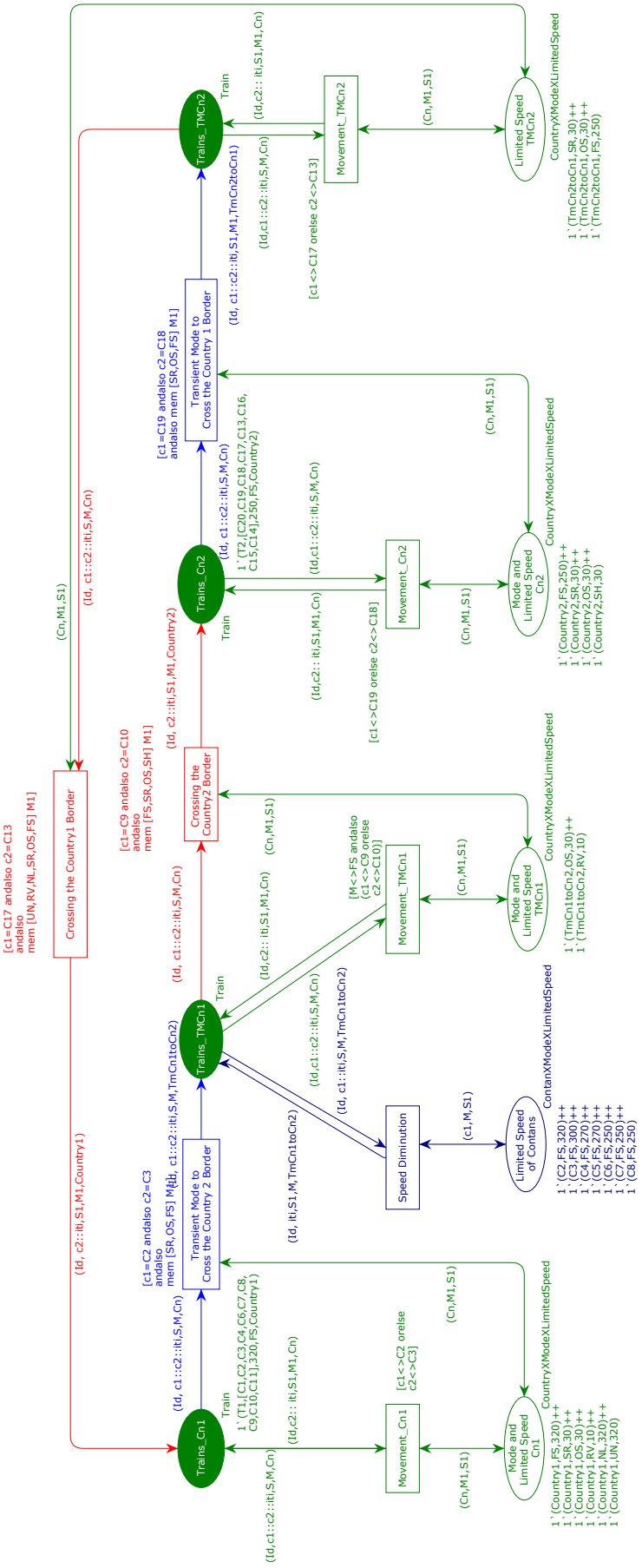



Figure 5.7 – Modèle du système ERTMS.

```
Statistics
------------------------------------------------------------------

 State Space                      Scc Graph

    Nodes:  1260                    Nodes:  1260
    Arcs:   7133                    Arcs:   7133
    Secs:   0                       Secs:   0
    Status: Full

Boundedness Properties
------------------------------------------------------------------

  Best Integer Bounds                 Upper      Lower
     Limited_Speed_TMCn2               3          3
     Limited_Speed_of_Contans          7          7
     Mode_and_Limited_Speed_Cn1        6          6
     Mode_and_Limited_Speed_Cn2        4          4
     Mode_and_Limited_Speed_TMCn1      2          2
     Trains_Cn1                        2          0
     Trains_Cn2                        2          0
     Trains_TMCn1                      1          0
     Trains_TMCn2                      1          0

  Best Upper Multi-set Bounds
     Trains_Cn1 1
1`(T1,[C1,C2,C3,C4,C6,C7,C8,C9,C10,C11],320,FS,Country1)++
1`(T1,[C2,C3,C4,C6,C7,C8,C9,C10,C11],10,RV,Country1)++
1`(T1,[C2,C3,C4,C6,C7,C8,C9,C10,C11],30,SR,Country1)++
1`(T1,[C2,C3,C4,C6,C7,C8,C9,C10,C11],30,OS,Country1)++
1`(T1,[C2,C3,C4,C6,C7,C8,C9,C10,C11],320,FS,Country1)++
1`(T1,[C2,C3,C4,C6,C7,C8,C9,C10,C11],320,UN,Country1)++
1`(T1,[C2,C3,C4,C6,C7,C8,C9,C10,C11],320,NL,Country1)++
1`(T2,[C13,C16,C15,C14],50,SR,Country1)++
1`(T2,[C13,C16,C15,C14],50,OS,Country1)++
1`(T2,[C13,C16,C15,C14],250,FS,Country1)++
1`(T2,[C14],10,RV,Country1)++
1`(T2,[C14],30,SR,Country1)++
1`(T2,[C14],30,OS,Country1)++
1`(T2,[C14],320,FS,Country1)++
1`(T2,[C14],320,UN,Country1)++
1`(T2,[C14],320,NL,Country1)++
1`(T2,[C15,C14],10,RV,Country1)++
1`(T2,[C15,C14],30,SR,Country1)++
1`(T2,[C15,C14],30,OS,Country1)++
1`(T2,[C15,C14],320,FS,Country1)++
1`(T2,[C15,C14],320,UN,Country1)++
1`(T2,[C15,C14],320,NL,Country1)++
1`(T2,[C16,C15,C14],10,RV,Country1)++
1`(T2,[C16,C15,C14],30,SR,Country1)++
1`(T2,[C16,C15,C14],30,OS,Country1)++
1`(T2,[C16,C15,C14],320,FS,Country1)++
1`(T2,[C16,C15,C14],320,UN,Country1)++
1`(T2,[C16,C15,C14],320,NL,Country1)
```

```
     Trains_TMCn1 1
1`(T1,[C2,C3,C4,C6,C7,C8,C9,C10,C11],10,RV,TmCn1toCn2)++
1`(T1,[C2,C3,C4,C6,C7,C8,C9,C10,C11],30,SR,TmCn1toCn2)++
1`(T1,[C2,C3,C4,C6,C7,C8,C9,C10,C11],30,OS,TmCn1toCn2)++
1`(T1,[C2,C3,C4,C6,C7,C8,C9,C10,C11],320,FS,TmCn1toCn2)++
1`(T1,[C2,C3,C4,C6,C7,C8,C9,C10,C11],320,UN,TmCn1toCn2)++
1`(T1,[C2,C3,C4,C6,C7,C8,C9,C10,C11],320,NL,TmCn1toCn2)++
1`(T1,[C3,C4,C6,C7,C8,C9,C10,C11],10,RV,TmCn1toCn2)++
1`(T1,[C3,C4,C6,C7,C8,C9,C10,C11],30,OS,TmCn1toCn2)++
1`(T1,[C3,C4,C6,C7,C8,C9,C10,C11],320,FS,TmCn1toCn2)++
1`(T1,[C4,C6,C7,C8,C9,C10,C11],10,RV,TmCn1toCn2)++
1`(T1,[C4,C6,C7,C8,C9,C10,C11],30,OS,TmCn1toCn2)++
1`(T1,[C4,C6,C7,C8,C9,C10,C11],300,FS,TmCn1toCn2)++
1`(T1,[C6,C7,C8,C9,C10,C11],10,RV,TmCn1toCn2)++
1`(T1,[C6,C7,C8,C9,C10,C11],30,OS,TmCn1toCn2)++
1`(T1,[C6,C7,C8,C9,C10,C11],270,FS,TmCn1toCn2)++
1`(T1,[C7,C8,C9,C10,C11],10,RV,TmCn1toCn2)++
1`(T1,[C7,C8,C9,C10,C11],30,OS,TmCn1toCn2)++
1`(T1,[C7,C8,C9,C10,C11],250,FS,TmCn1toCn2)++
1`(T1,[C8,C9,C10,C11],10,RV,TmCn1toCn2)++
1`(T1,[C8,C9,C10,C11],30,OS,TmCn1toCn2)++
1`(T1,[C8,C9,C10,C11],250,FS,TmCn1toCn2)++
1`(T1,[C9,C10,C11],10,RV,TmCn1toCn2)++
1`(T1,[C9,C10,C11],30,OS,TmCn1toCn2)++
1`(T1,[C9,C10,C11],250,FS,TmCn1toCn2)

     Trains_Cn2 1
1`(T1,[C10,C11],30,OS,Country2)++
1`(T1,[C11],50,SH,Country2)++
1`(T1,[C11],50,SR,Country2)++
1`(T1,[C11],50,OS,Country2)++
1`(T1,[C11],250,FS,Country2)++
1`(T2,[C19,C18,C17,C13,C16,C15,C14],50,SH,Country2)++
1`(T2,[C19,C18,C17,C13,C16,C15,C14],50,SR,Country2)++
1`(T2,[C19,C18,C17,C13,C16,C15,C14],50,OS,Country2)++
1`(T2,[C19,C18,C17,C13,C16,C15,C14],250,FS,Country2)++
1`(T2,[C20,C19,C18,C17,C13,C16,C15,C14],250,FS,Country2)

     Trains_TMCn2 1
1`(T2,[C17,C13,C16,C15,C14],50,SR,TmCn2toCn1)++
1`(T2,[C17,C13,C16,C15,C14],50,OS,TmCn2toCn1)++
1`(T2,[C17,C13,C16,C15,C14],250,FS,TmCn2toCn1)++
1`(T2,[C18,C17,C13,C16,C15,C14],50,SR,TmCn2toCn1)++
1`(T2,[C18,C17,C13,C16,C15,C14],50,OS,TmCn2toCn1)++
1`(T2,[C18,C17,C13,C16,C15,C14],250,FS,TmCn2toCn1)++
1`(T2,[C19,C18,C17,C13,C16,C15,C14],50,SR,TmCn2toCn1)++
1`(T2,[C19,C18,C17,C13,C16,C15,C14],50,OS,TmCn2toCn1)++
1`(T2,[C19,C18,C17,C13,C16,C15,C14],250,FS,TmCn2toCn1)
```

```
 Home Properties
-------------------------------------------------------------

  Home Markings
     None


 Liveness Properties
-------------------------------------------------------------

   Dead Markings
      24 [1260,1259,1258,1257,1256,...]

   Dead Transition Instances
      None

   Live Transition Instances
      None
```

Figure 5.8 – Extrait du rapport d'espace d'états.

Dans ce qui suit, nous allons nous intéresser à la vérification des propriétés importantes dans le comportement du système ERTMS qui sont :

**Mode actif :** chaque train ne peut être que dans un et un seul mode de fonctionnement à la fois.

D'après Définition 29, un mode opératoire appliqué aux SdSs est un élément du produit cartésien de l'ensemble des trains par l'ensemble des systèmes sol. Chaque système sol peut avoir lui même plusieurs modes sol.

D'après le rapport, la meilleure borne entière supérieure des places $Trains\_Cn1$ et $Trains\_Cn2$ est 2 alors que celle de $Trains\_TMCn1$ et $Trains\_TMCn2$ est 1. Ceci signifie d'une part qu'il existe au plus deux jetons dans les places $Trains\_Cn1$ et $Trains\_Cn2$ et qu'il existe des marquages accessibles où il existe deux jetons ; et d'autre part, qu'il existe au plus un jeton dans les places $Trains\_TMCn1$ et $Trains\_TMCn2$ et qu'il existe des marquages accessibles où il existe un jeton.

D'un autre côté, la meilleure borne entière inférieure des places citées ci-dessus est 0, ce qui signifie que ces places peuvent être vide.

En effet, ces résultats prouvent que les deux trains de notre exemple changent bien de modes. Il faut maintenant vérifier la couleur des jetons de ces 4 places. Nous considérons alors les meilleurs multi-ensembles supérieurs des bornes des places $Trains\_Cn1$, $Trains\_Cn2$, $Trains\_TMCn1$ et $Trains\_TMCn2$. D'après le rapport, le jeton modélisant $Tr1$ est bien dans la place $Trains\_Cn1$ uniquement quand le train est sur le segment $C1$. Ensuite nous le trouvons dans la place $Trains\_TMCn1$ du mode transitoire quand le train traverse les segments $C2$, $C3$, $C4$, $C6$, $C7$, $C8$ et $C9$. Dans la place $Trains\_Cn2$, le jeton de $Tr1$ ne s'y trouve que quand le train

traverse $C10$ et $C11$. À partir de ce constat, nous pouvons en conclure qu'un seul mode de fonctionnement est activé à la fois pour $Tr1$.
Le même résultat s'applique aussi pour $Tr1$ qui a la trajectoire $[C20, C19, C18, C17, C13, C16, C15, C14]$ car il est dans $Trains\_Cn2$ quand il traverse $C20$, dans $Trains\_TMCn2$ quand il traverse $C17$, $C18$ et $C19$ et dans $Trains\_Cn1$ quand il traverse $C13$, $C14$, $C15$ et $C16$.

**Arrêt des trains :** Les trains ne s'arrêtent que s'ils ont terminé leurs missions.

Notre modèle contient 24 marquages morts selon le rapport d'espace d'état. Un marquage mort est un état du réseau dans lequel aucune transition n'est activée. Ces marquages morts sont bien causés par les deux trains qui sont arrivés à la fin de leurs trajectoires. Notre modèle fait valoir que lorsque les trains sont à destination, rien d'autre ne peut arriver. Dans ce cas, les jetons situés dans $Trains\_Cn1$ et $Trains\_Cn2$ ne peuvent pas être supprimés et forment un marquage mort.

L'espace d'état indique les états qui sont des marquages morts. Par exemple, l'état «1260» de l'espace d'états est un marquage mort. En utilisant l'outil *Display the node with the specified number* du CPN Tools, nous pouvons visualiser cet état comme le montre la Figure 5.9. Il correspond à l'état où le train $Tr1$ est à sa destination $C11$ en mode opératoire $SH$ et $Tr2$ est au segment $C14$ en mode opératoire $NL$. En effet, cet état est un marquage mort valide.

De plus, les trains circulent pour atteindre leurs destinations. Dans notre cas, il n'est généralement pas souhaitable de revenir à un état antérieur dans le cadre d'une circulation commerciale. Ainsi le modèle ne devrait pas avoir de marquage d'accueil (*home-marking*) et l'analyse d'espace d'état montre que c'est effectivement le cas.

## 5.6 Conclusion

Ce chapitre a formalisé le passage des frontières sous ERTMS Niveau 2 en le considérant comme un SdS et en utilisant la problématique de gestion de mode de fonctionnement connue dans la littérature des SEDs. En effet, le système ERTMS est divisé en plusieurs modes de fonctionnement, en utilisant une approche multi-modèle. Chaque mode de fonctionnement représente l'interaction entre un système sol et un train. Ce chapitre présente une approche fournissant une vision globale de la gestion des comportements des trains sur l'espace européen, au lieu de plusieurs vues nationales, tout en proposant une méthode de gestion sûre des modes de fonctionnement lorsque les trains franchissent les frontières. Cette approche apporte un complément des spécifications comportementales de la relation entre les automatismes à bord des trains gérés par le European Vital computer (EVC) et le Radio Block Center (RBC), recevant les informations provenant du sol, pour franchir

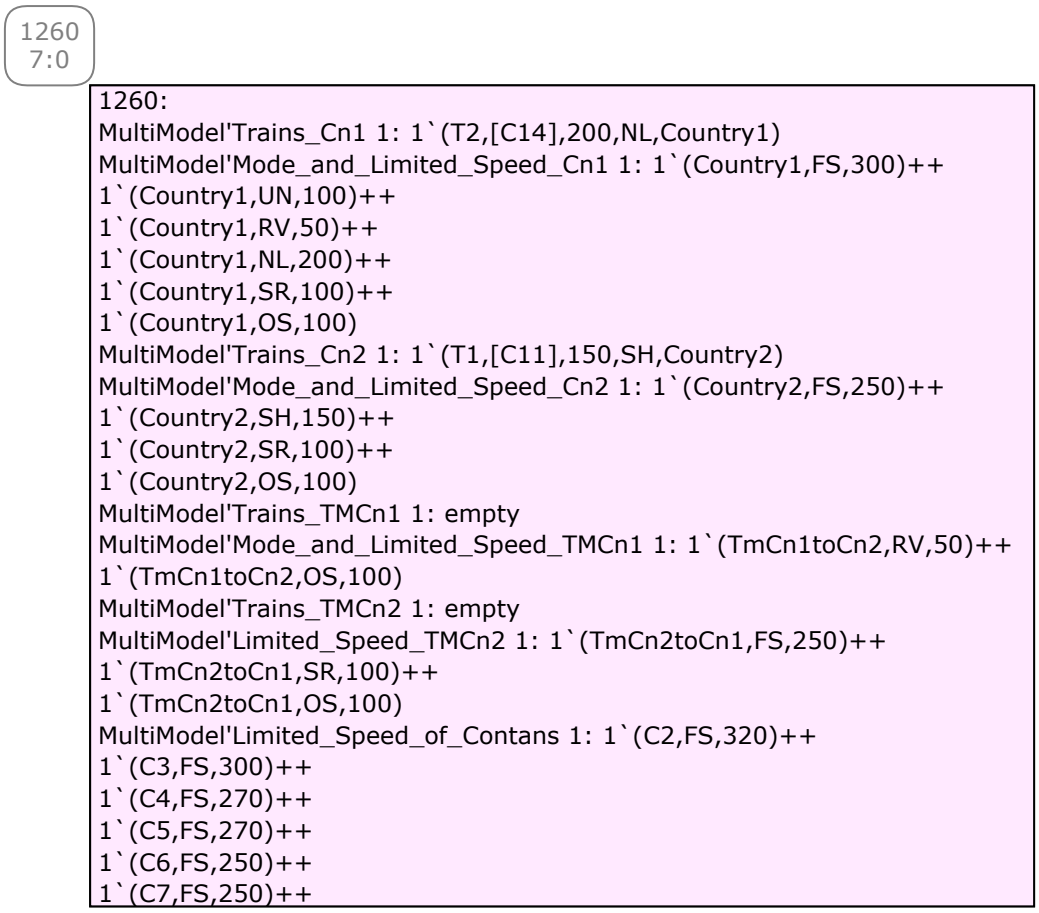



Figure 5.9 – Etat «1260»

une frontière.

# Chapitre 6

# Systèmes de gestion de crise

## 6.1 Introduction

Les Infrastructures Critiques (ICs) sont des infrastructures importantes pour le bien-être et même pour l'existence des sociétés. Il est donc primordial de veiller à ce que les ICs fonctionnent correctement. Si un certain type d'événement imprévu est inéluctable, il est de la plus haute importance d'être préparé à sa gestion afin d'assurer un retour rapide à l'état de fonctionnement de l'IC. Ce chapitre présente deux approches originales pour la gestion des crises des ICs. La première approche traite les crises émergentes et la deuxième les crises frontalières. Les approches proposées reposent sur la gestion des modes de fonctionnement de la TCS. L'applicabilité et la généralité de chacune des approches proposées est démontrée dans une étude de cas.
Ce chapitre s'appuie sur des travaux réalisés dans le cadre d'un projet franco-allemand RE(H)STRAIN. Les résultats présentés dans ces lignes sont issue d'une collaboration avec TH Köln-Université des sciences appliquées.

## 6.2 Gestion de crise

L'Union Internationale des Chemins de fer (UIC) définit une crise comme un événement soudain ou un ensemble de circonstances pouvant affecter de manière significative la capacité d'une organisation à mener ses activités ou des événements qui pourrait avoir un impact négatif sur la réputation d'une organisation, avoir des conséquences néfastes pour l'environnement ou le grand public [UIC-Security Division 2017]. Selon la British Standards Institution (BSI), la gestion de crise fait généralement référence au développement et à l'application de la capacité organisationnelle à traiter, ou à limiter autant que faire ce peut, les conséquences négatives des événements [BS 11200:2014]. Habituellement, les stratégies et les actions respectives incluent la préparation de potentiels événements à venir

ainsi que la gestion des incidents imprévus. Ils sont consignés dans un Plan de Gestion de Crise (PGC). Le PGC peut être considéré comme un document de haut- niveau qui fournit des directives générales organisationnelles et procédurales pour la gestion de l'information, des activités, des opérations et des communications en cas d'urgence [UIC-Security Division 2017]. Il est préconisé, en cas de crise, de créer une structure organisationnelle dédiée capable de faire face à aux problèmes résultants de manière appropriée et efficace ; l'équipe de gestion de crise (EGC) fait souvent partie de cette structure [UIC-Security Division 2017, BS 11200:2014]. Les défis auxquels la gestion de crise est généralement confrontée sont les suivants : niveaux d'incertitude élevés, temps limité, informations limitées et ressources limitées. Dans les zones frontalières ou lorsqu'une coopération inter-organisations est requise, ces défis sont amplifiés. Selon [Papatheodorou et al. 2014], qui a analysé les activités antérieures de l'UE en matière de mitigation des conséquences des catastrophes naturelles, le manque de capacités et de ressources pour la coopération, la connaissance limitée des organisations coopérantes et les différences de structures et de procédures organisationnelles sont des problèmes récurrents dans la coopération transfrontalière. Ces problèmes doivent être résolus ou contournés car, en raison de l'évolution du paysage des menaces, la coopération transfrontalière est de plus en plus importante pour la gestion de crise [Boin et al. 2014, Surminski et al. 2017].
Cette section décrit les grand enjeux de la gestion de crise. En outre, des recommandations visant à améliorer la réaction aux crises surtout dans les zones frontalières seront formulées sur la base des normes et directives existantes. De plus, les résultats d'une enquête effectuées auprès d'experts en gestion de crise réalisée au cours du projet RE(H)STRAIN.

### 6.2.1 Gestion de crise et infrastructure critique

La gestion de crise est pertinente dans tous les types d'organisations : les moyennes entreprises ainsi que les sociétés opérant au niveau mondial et les départements/autorités gérés par l'État ainsi que les organisations non-gouvernementales (ONGs) et le grand public. Dans certaines zones, la gestion de crise revêt une importance plus grande en raison du grand nombre d'incidents potentiellement négatifs, y compris pour des tiers tels que l'environnement, d'autres entreprises ou le grand public. Les ICs sont un exemple pour une telle zone. [Ministère fédéral de l'intérieur de l'Allemagne 2009] définit une IC comme une infrastructure si importante pour le bon fonctionnement de la société qu'en cas de panne, des pénuries persistantes d'approvisionnement, des perturbations importantes de la sûreté et de la sécurité publiques, ou d'autres dramatiques conséquences pourrait se produire. Un exemple d'IC est le transport public - dans le cadre de ce chapitre, nous nous concentrons sur le transport ferroviaire de voyageurs. Ce transport public se compose d'un réseau transfrontalier complexe de trains, de voies et de gares et il joue un rôle majeur

dans le bon fonctionnement des sociétés modernes [Lévy-Bencheton et Darra 2015].
En raison de son importance pour la société et de la complexité de son réseau, les transports publics devraient se préparer aux crises afin que les fonctions fondamentales de l'infrastructure soient au maximum préservées et rétablies le plus rapidement possible. Ainsi, les organisations du secteur des transports en commun devraient établir leur propre gestion de crise. Les normes et les directives sont des outils précieux pour la mise en place d'une structure de gestion de crise.

### 6.2.2 Gestion de crise dans les transports en commun

Un élément clé d'une telle structure est le PGC, qui comprend des procédures fixes à suivre en cas de crise. Le contenu précis du PGC doit être déterminé individuellement car la gestion de crise varie d'une organisation à l'autre et d'un secteur à l'autre [BS 11200:2014]. En conséquence, toutes les organisations devraient créer leur propre PGC adapté à leurs besoins. Mais comme il existe des exigences générales en matière de gestion de crise indépendantes du secteur concerné, les normes peuvent constituer une aide importante pour la mise en place d'un PGC. De plus, les normes et les directives constituent une source qui peut être utilisée pour déterminer l'état de l'art actuel de la gestion de crise ciblée. Afin de recueillir des informations complètes sur l'état de l'art actuel de la gestion de crise dans les transports en commun, en mettant l'accent sur la gestion de crise transfrontalière, trois sources différentes ont été exploitées : les normes et les directives existantes sont analysées, une analyse documentaire des projets de recherche en cours est menée et une enquête internationale anonyme est distribuée aux responsables de la gestion de crise des entreprises ferroviaires. Dans la section suivante, les sources et les résultats obtenus, les informations recueillies seront brièvement présentées.

### 6.2.3 Normes et directives de la gestion de crise

Au total, quatre normes, largement acceptées et d'une grande pertinence, ont été étudiées dans cette section. [BS 11200:2014] ainsi que [UIC-Security Division 2017] sont abordés dans ce travail. [ISO 22320:2018] et [Bundesamt für Verfassungsschutz 2008] (Allemagne) sont également de précieuses sources d'information mais ne sont pas explicitement mentionnées dans le présent document.

La norme [BS 11200:2014] est largement acceptée. Elle est destinée à toute organisation, quel que soit son emplacement, sa taille, son type, son industrie ou son secteur. Le document fournit des indications précises pour comprendre la gestion de crise. Il aide à établir une structure de gestion de crise et à préparer le EGC aux défis auxquels il devra faire face et il fournit des informations sur la communication appropriée en cas de crise.

En outre, la question de la formation, des nécessaires exercices et de l'apprentissage des crises est explicitement abordée.

La norme [UIC-Security Division 2017] a été élaborée sur la base d'entretiens avec des experts, d'enquêtes et de retours d'expériences. Elle consiste toutefois à conseiller les opérateurs ferroviaires pour les aider, entre autres, à élaborer un PGC, à mettre en place une Équipe de Gestion de Crise (EGC) compétante, à comprendre la définition des acteurs clés et de leurs fonctions, à estimer l'infrastructure qui devrait être fournie afin de permettre une gestion efficace de crise et, en outre, à instruire les opérateurs sur les bases de la communication de crise. Les autres sujets abordés sont la formation, l'évaluation et la mise à jour des structures de gestion de crise, l'établissement des priorités et l'établissement des niveaux d'alerte.

Les deux normes ont plutôt un caractère indicatif et fournissent des conseils pratiques aux hauts dirigeants dans le cas de la [BS 11200:2014] et aux opérateurs ferroviaires dans le cas de la [UIC-Security Division 2017]. Il est à noter que la norme [UIC-Security Division 2017] est légèrement plus détaillée, ce qui est probablement dû au fait qu'elle se concentre exclusivement sur le transport ferroviaire.
Comme ces deux normes ont été publiées récemment, elles peuvent être utilisées pour identifier les meilleures pratiques, de l'état de l'art actuel de la gestion de crise. Les éléments des deux normes sont assez similaires et suggèrent que les aspects suivants peuvent être considérés comme de bonnes pratiques et devraient être mis en oeuvre dans les organisations pour assurer une gestion efficace des situations de crise :

1. Procédures documentées et fixes pour la gestion des informations, des opérations, etc. en cas de crise avec différents niveaux d'escalade (développer un PGC),
2. Nommer les personnes responsables des opérations respectives (les décideurs) et clarifier l'étendue de leurs responsabilités (établir un EGC),
3. Définir les principes de communication en cas de crises et
4. Mener des entraînements et des exercices pour développer une pratique de ce qui précède.

Pour compléter ces bonnes pratiques, les projets de recherche en cours traitant (au moins en partie) de la gestion de crise sont analysés pour identifier les défis réels et les recommandations futures.

### 6.2.4 Projets de recherche

Actuellement, de nombreux projets de recherche traitent la gestion de crise, même s'ils ne constituent généralement qu'un élément d'un effort visant à améliorer la résilience ou la sécurité. Le fait que la gestion de crise contribue à améliorer la résilience est également reconnu dans cette norme relativement récente [ISO 22316:2017]. Les domaines de

recherche individuels ainsi que les résultats des projets de recherche donnent des indications sur l'importance des composantes individuelles de la gestion de crise. Notamment, elle instruit sur les faiblesses de la gestion de crise.
Pour le présent document, une sélection de six projets de recherche financés par la Commission européenne est brièvement présentée. Elle est ensuite prise en compte pour l'élaboration de recommandations. Le tableau 6.1 présente un aperçu des projets de recherche en cours sur la gestion des crises transfrontalières.
L'aperçu n'est pas exhaustif, mais décrit simplement une sélection censée fournir des informations sur les domaines de la gestion de crise récemment examinés. Comme mentionné précédemment, la gestion de crise impliquant une coopération multinationale n'est le plus souvent qu'un élément d'une étude plus vaste. Bien que les résultats puissent parfois sembler similaires, toutes ces approches de recherche ont un objectif différent. Pendant que FORTRESS et PREDICT se concentrent sur les effets de cascade, EPISECC, CIPRNet et PREPARE mettent principalement l'accent sur le partage d'informations. START contribue à la gestion de crise en mettant en place les structures respectives au niveau régional.

Outre les projets de recherche brièvement présentés, qui sont principalement financés par la Commission européenne, de nombreuses autres actions visent à améliorer la préparation aux catastrophes ou à améliorer la réaction en harmonisant les procédures et la sémantique (e.g. [Vademecum – Civil Protection]), par le biais du partage d'informations (e.g. [DRMKC, UN-SPIDER]) ou en fournissant des capacités de coordination et de déploiement de services d'urgence (e.g. [ERCC, EERC]).
Des critiques ont toutefois été formulées concernant la capacité de coordination et l'efficacité des initiatives de l'UE dans le contexte de la gestion de crises transfrontalières [Boin et al. 2014].

Les normes existantes et les projets de recherche en cours fournissent déjà de bonnes informations sur l'état de l'art de la gestion des crises et sur les défis les plus importants de ce contexte. Toutefois, la revue de la littérature a montré qu'il n'existe ni de projet de standard ni de projet de recherche exclusivement axé sur la gestion de crise transfrontalière.

### 6.2.5 Enquête auprès des gestionnaires de crise internationaux

Afin de recueillir des informations de première main sur la coopération transfrontalière en cas de crise, une enquête a été réalisée. L'objectif est de rassembler des informations sur l'état actuel de la gestion de crise dans les entreprises, les pratiques actuelles en matière de coopération transfrontalière et les tendances futures. Les questions sont axées sur les partenariats transfrontaliers, les exercices et la formation en commun, la communication et le partage des connaissances et le potentiel d'optimisation.

| | CIPRNet | EPISECC | FORTRESS | PREDICT | PREPARE | START |
|---|---|---|---|---|---|---|
| **Sector** | Information et Communication | Information et Communication | Interdisciplinaire | Interdisciplinaire | Services de santé | Protection Civil |
| **Sujet principal** | Protection des infrastructures critiques | «Leçons apprises» - crises européennes passées | Effets en cascade | Effets en cascade/Protection des infrastructures critiques | Prévention de la pandémie | Réponse et prévention des inondations |
| **Résultat principal** | Plateforme multiple d'échanges entre chercheurs et praticiens. Développement d'une simulation de modélisation «quel-si» et d'une capacité d'analyse permettant aux décideurs de prédire les conséquences de leurs actions. | L'analyse des réponses passées aux crises et la comparaison des structures organisationnelles des gestionnaires de crise et des premiers intervenants sont les résultats d'EPISECC. Sur cette base, un inventaire détaillé contenant les crises passées et leur traitement est élaboré. Les procédures opératoires standard sont identifiées. | Recommandations pour la gestion de crise lors des événements transfrontaliers mettant l'accent sur la communication et la sémantique. De plus, Fiet-un logiciel qui prédit les conséquences des actions et aide ainsi les décideurs, a été développé. | PREDICT fournit des méthodologies qui facilitent la formation des responsables de crise en mettant un accent particulier sur les effets en cascade et permettent aux utilisateurs finaux d'auto-évaluer le risque de leur organisation afin de déclencher ou être affecté par des effets en cascade spécialement pour les infrastructures critiques. | PREPARE est en train de développer des plates-formes susceptibles de contribuer et de faciliter le partage d'informations. Ces plateformes ont accès à de grands réseaux et pourraient donc être utilisées pour identifier et déployer les experts appropriés et lancer rapidement des études de recherche clinique. | Développement d'un plan d'intervention commun, mise en place d'une EGC transfrontalière et création d'une base de données permettant une vue d'ensemble des ressources locales. Les services d'urgence locaux et les volontaires sont formés aux nouvelles structures et procédures. |

TABLE 6.1 – Projets de recherche traitant la gestion de crise.

Grâce au soutien de l'UIC, l'enquête a pu être distribuée à de nombreux services de gestion de crise des opérateurs ferroviaires. Au total, 16 experts en gestion de crise ont répondu à

l'enquête. Les participants venaient d'Europe, d'Amérique du Nord, d'Asie et du Moyen-Orient. En raison de l'anonymat de l'enquête et du nombre limité de participants, les résultats obtenus doivent être interprétés avec une extrême prudence.
Lorsque nous avons comparé les réponses des experts, il est devenu évident que leurs opinions sur le niveau de leur implication dans la coopération transfrontalière sont loin d'être homogène. Par exemple, sur douze experts qui ont répondu aux questions, sur l'existence d'une procédure d'alarme normalisée auprès des entreprises étrangères et sur la connaissance de la structure de l'organisation des entreprises étrangères en cas de crise, dans les deux cas, six ont répondu « oui » et six « non » (voir Figure 6.1).
Ce même cas se reproduit pour plusieurs questions concernant la coopération et les pro-

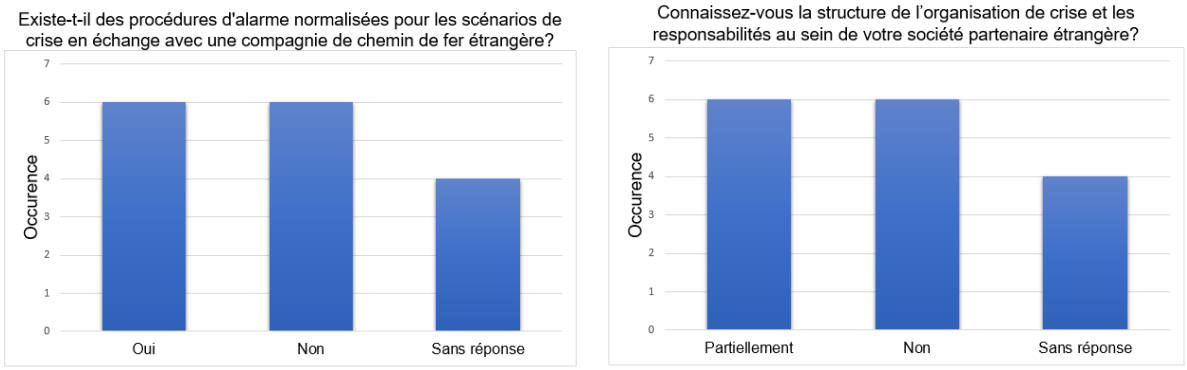



Figure 6.1 – Résultats de l'enquête concernant les procédures d'alarme normalisées et la connaissance de la structure organisationnelle.

cédures conjointes. En conséquence, les experts sont répartis de manière égale quant à leur utilisation des normes/directives pour la préparation de la gestion de crise transfrontalière. Cinq participants prennent en considération de telles normes, six autres non. La Figure 6.2 montre cependant que presque tous les experts considèrent que la gestion de crise transfrontalière comporte deux choses : des exercices conjoints et des programmes d'échange destinés aux responsables de la gestion de crise. Ces résultats contribuent à élaborer des recommandations pour la gestion de crise transfrontalière. Entre autres, il est recommandé :

— d'organiser des exercices communs réguliers,

— d'établir un cadre commun de communication, y compris des protocoles d'alerte avec des sociétés partenaires étrangères et

— de faire connaissance avec les autres structures organisationnelles en situation de crise, éventuellement par le biais d'un échange de responsables de crise.

Une autre conclusion intéressante dérivée des réponses en texte libre contredit les efforts déployés pour harmoniser les structures organisationnelles et les procédures de réaction aux crises au niveau international.

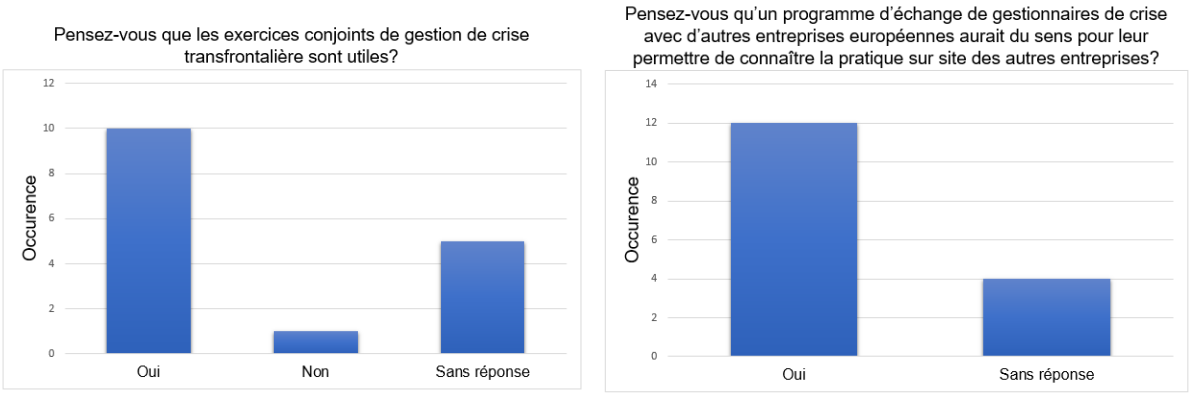



Figure 6.2 – Résultats de l'enquête sur les exercices conjoints et les programmes d'échange pour les gestionnaires de crise.

Bien que la plupart des gestionnaires de crise (six sur onze) déclarent qu'ils considéreraient un guide uniforme pour la gestion de crise transfrontalière utile à leur travail, quatre experts ont exprimé leur désaccord, précisant que des différences d'organisation et de réglementation entre les États sont trop étendues pour être harmonisées dans une seule directive. Dans le cadre du projet de recherche [EPISECC], les différences d'approches de différents pays ont été identifiées, les procédures de gestion de crise et les normes de différents pays européens étant collectées et comparées. Cela suggère qu'une autre approche, au lieu de poursuivre l'harmonisation, pourrait être plus prometteuse. Dans la section 6.4, un cadre formel, gérant les coopérations internationales en respectant les différences d'organisation et de réglementation entre les États, est présenté. Mais nous traitons d'abord, dans la section suivante, le problème des séries de crises dans un même État.

## 6.3 Gestion des séries de crise

Bien que chaque situation de crise soit unique, les réponses adaptées partagent des éléments communs. Ainsi, plusieurs plans de crise prédéfinis dans le PGC peuvent être appliqués en fonction de l'ampleur de la crise et des dommages, matériels et corporels, qui en résultent.

Un plan de crise rassemble l'ensemble des données et ressources nécessaires pour répondre à l'urgence de la situation dans un périmètre géographique précis.

Par ailleurs, lorsque la gestion de crise est requise, la réponse est coordonnée dans le cadre du Système de Gestion de Crise (SGC). Ce système doit être pro-actifs et prendre en compte les possibilités d'évolution de la crise et les éventualités d'incidents émergents. Un changement de plan de crise doit donc être rationnellement envisagé dont le but de cette section. En effet, et en appliquant avec adaptation la démarche proposée dans la Section 3,

nous proposons une approche de modélisation du SGC garantissant une commutation sûr entre les plans et qui résout le problème des séries de crise par fusion des plans.
Le fonctionnement de cette approche est le suivant : étant donné un plan de crise déclenché dans un ensemble de zones données. Ces zones, appelées zones de crise, mettent à disposition du décideur toutes leurs ressources humaines et matérielles. Lors de l'apparition d'une nouvelle crise dans une zone de crise, deux cas de figure se présentent. Dans le premier cas, la nouvelle crise se situe dans la même zone sinistrée par la première crise. Dans ce cas, les deux crises sont vues comme une seule tout en gardant la possibilité au décideur d'augmenter le niveau du plan de crise activé par un autre couvrant plus de zones pour mettre à disposition plus de ressources. Dans le deuxième cas, la nouvelle crise se produit dans une zone de crise différente de la zone sinistré (voir, à titre d'exemple, la Figure 6.3). Alors un nouveau plan de crise approprié au degré de la nouvelle crise est déclenché dans la zone sinistrée et une fusion entre les deux plans de crise s'effectue juste après. Le nouveau plan ainsi construit met à disposition du décideur toutes les ressources, humaines et matérielles, issue de l'union des zones de crises des deux premiers plans activés en amont.

### 6.3.1 Modélisation du SGC

Un SGC est un système dynamique constitué de plusieurs plans de crise qui se déclenchent et se fusionnent en réponse aux crises émergentes.

**Définition 32 (Ensemble des plans de crise)**
*L'ensemble des plans de crise est appelé* $Pl = \{pl_1, pl_2, ..., pl_{|Pl|}\}$ *où* $|Pl| \geq 1$.

**Définition 33 (Ensemble des zones)**
*L'ensemble des zones est appelé* $A = \{a_1, a_2, ..., a_{|A|}\}$ *où* $|A| \geq 1$.

**Définition 34 (Plan de crise activé)**
*L'ensemble des plans de crise activés est appelé* $PL$ *où*

$$\begin{array}{rcl} A \times Pl & \longrightarrow & PL \\ (a_i, pl_j) & \longmapsto & PL_{i,j} \end{array}$$

*est une application tel que* $PL_{i,j}$ *indique le plan* $pl_j$, $j = 1..|Pl|$, *activé dans la zone sinistrée* $a_i$, $i = 1..|A|$.

**Définition 35 (Zones de crises)**
*Soit* $PL_{i,j}$ *un plan de crise activé où* $i \in 1..|A|$ *et* $j \in 1..|Pl|$. *L'ensemble des zones de crise concernés par* $PL_{i,j}$ *est appelé* $A_{i,j}$.

Dans ce qui suit, le SGC est vue comme un SED pour pouvoir se baser sur le concept de gestion des modes de fonctionnement abordé dans l'état de l'art. Un mode de fonctionnement du SGC est alors un plan de crise activé. Pour chaque zone sinistrée, une commutation est possible vers d'autres plans de crise de couverture plus étendue ou construits par fusion de plans. La gestion des modes de fonctionnement assume aussi qu'un seul mode de fonctionnement est actif à la fois pour toute zone sinistrée simple ou composée.
De plus, nous adoptons l'approche multi-modèle, dans laquelle un modèle RdPCP différent est associé à chaque mode de fonctionnement. Ceci permet de définir un comportement distinct et une stratégie de contrôle différente pour chaque mode de fonctionnement.
À Chaque plan de crise activé $PL_{i,j}$, $i \in 1..|A|$ et $j \in 1..|Pl|$, nous associons un modèle RdPCP $< P_{i,j}, T_{i,j}, K_{i,j}, D_{i,j}, W^-_{i,j}, W^+_{i,j}, \phi_{i,j}, M_{0,i,j}, \Pi_{i,j} >$.
Pour des raisons de simplicité, la même identité du plan activé $PL_{i,j}$ est attribuée à son RdPCP associé.

### Modèle abstrait de plan de crise

Un même plan de crise peut être activé/désactivé simultanément dans plusieurs zones. Par conséquent, le même comportement apparaît dans tout les plans de crise activés associés, ce qui nous amène à définir le concept de modèle abstrait de plan de crise.

**Définition 36 (Modèle abstrait de plan de crise)**
*Un modèle RdPCP* $< P_j, T_j, K_j, D_j, W^-_j, W^+_j, \phi_j, M_{0,j}, \Pi_j >$ *est un modèle abstrait associé au plan de crise* $Pl_i$ *si, et seulement si,* $\forall PL_{i,j} =< P_{i,j}, T_{i,j}, K_{i,j}, D_{i,j}, W^-_{i,j}, W^+_{i,j}, \phi_{i,j}, M_{0,(i,j)}, \Pi_{i,j} >$ *où* $i \in \{1..|A|\}$,

$$< P_j, T_j, K_j, D_j, W^-_j, W^+_j, \phi_j, \Pi_j >=< P_{i,j}, T_{i,j}, K_{i,j}, D_{i,j}, W^-_{i,j}, W^+_{i,j}, \phi_{i,j}, \Pi_{i,j} >$$

$$et\ M_{0,i,j} \subseteq M_{0,j}.$$

**Remarque 6**

- *Pour des raisons de simplicité, la même identité du plan activé* $Pl_i$ *est attribuée à son modèle RdPCP abstrait associé.*
- *Par convention, tout* $Pl_i$ *doit intégrer le mécanisme d'activation/désactivation des plans de crise.*

### Sous-modèle commun

Certains comportements similaires entre les plans de crise abstraits de niveau différent peuvent être soulignés surtout au niveau de gestion des ressources et, afin de minimiser la

taille du modèle global, un concept sous-modèle commun est défini.
Formellement, un sous-modèle commun est une partie du modèle RdPCP liée à deux ou plusieurs plan de crise abstrait différents.

**Définition 37 (Sous-modèle commun)**
$\forall(Pl_i, Pl_j) \in Pl \times Pl$ *où* $i \neq j$,
*si* $\exists c =< P_c, T_c, K_c, D_c, W_c^-, W_c^+, \phi_c, M_{0,c}, \Pi_c >$ *tel que* $(P_c = (P_i \cap P_j) \neq \emptyset)$ *ou* $(T_c = (Ti \cap T_j) \neq \emptyset)$ *alors c est un sous-modèle commun aux deux plans* $Pl_i$ *et* $Pl_j$ *si, et seulement si,*

*(a)* $\forall(p,t) \in P_c \times T_c, \quad W_i^- = W_j^-$ *et* $W_i^+ = W_j^+$,

*(b)* $\forall t \in T_c, \quad \phi_i = \phi_j$.

**Remarque 7** *De point de vue implémentation, le plan de crise abstrait mode de fonctionnement commun et le sous-modèle commun sont définis lorsque le concepteur utilise les même appellations dans les différents modèles RdPCPs.*

### Commutation de plan de crise

Un événement de commutation se déclenche entre deux plans de crise pour une même zone sinistrée. Le mécanisme de commutation provoque la désactivation du plan de crise actuel et l'activation d'un nouveau.
Le mécanisme de commutation est modélisé par des transitions RdPCP spécifiques dans chaque mode de fonctionnement. Pour distinguer ces transitions, une application est définie, dont le rôle est de fournir les informations du mode suivant.

**Définition 38 (Mécanisme de commutation)**
*Soit* $PL_{i,j}$, $i \in \{1..|A|\}, j \in \{1..|Pl|\}$ *un plan de crise activé et* $T'_{i,j} \subset T_{i,j}$ *un ensemble de transitions de commutation.*
*Soit* $Next\_plan : T'_{i,j} \rightarrow PL$ *une application tel que* $Next\_plan(t)$ *indique plan de crise à activer après le déclenchement de t,* $\forall t \in T'_{i,j}$.

### Fusion des plans de crise activés

La fusion des plans de crise activés est déclenchée automatiquement lorsqu'un nouveau plan de crise se déclenche et qu'un chevauchement de zones existe avec un autre plan de crise existant. Il est assuré par des transitions spécifiques de l'ensemble des transitions de commutation dont la priorité est plus élevée que les autres.

**Définition 39 (Mécanisme de fusion)**
*Soient* $PL_{i,j}$, $i \in \{1..|A|\}, j \in \{1..|Pl|\}$ *et* $PL_{k,l}$, $k \in \{1..|A|\}, l \in \{1..|Pl|\}$ *deux plans de*

*crise activés.*
*Si* $A_{i,j} \cap A_{k,l} \neq \emptyset$*, alors* $PL_{i,j}$ *et* $PL_{k,l}$ *sont remplacés par un nouveau plan de crise activé* $PL_{m,n}$ *tel que* $pl_n$ *est une liste dans laquelle* $pl_j$ *est concaténé à* $pl_l$ *et* $A_{m,n} = A_{i,j} \cup A_{k,l}$*.*

**Modèle de système de gestion de crise**

Sur la base des définitions précédentes, nous sommes maintenant en mesure de modéliser le SGC via un modèle global RdPCP à partir de plusieurs modèles représentant ses plans de crise activés.
Tout d'abord nous supposons que la place contenant les jetons représentants les plans de crise activés est appelée *Activated Plans*.
Le RdPCP global du SGC est un tuple $< P, T, K, D, W^-, W^+, \phi, M_0, \Pi >$ tel que :

$P = \cup_{(i=1..|Pl|)} P_i$;

$T = \cup_{(i=1..|Pl|)} T_i$;

$K = \cup_{(i=1..|Pl|)} K_i$;

$D$ est la fonction de domaine de couleur définie, par extension, à partir $P \cup T$ dans l'ensemble des domaines de couleur.

$W^-, W^+$ :

— $\forall Pl_i \in Pl, \forall (p,t) \in P_i \times T_i,$

$$W^-(p,t) = W_i^-(p,t),$$

$$W^+(p,t) = W_i^+(p,t);$$

— $\forall (Pl_i, Pl_j) \in Pl \times Pl \ (i \neq J),\ \forall (p,t) \in P_i \times T_j \ / p \notin P_j$ et $t \notin T_i,$

$$W^-(p,t) = W^+(p,t) = 0;$$

— $\forall Pl_i \in Pl,\ \forall t \in T'_j,$

$$W^+(Activated\ Plans, t) = Next\_plan(t);$$

où $T'_i$ est l'ensemble des transitions de commutation tel que $T'_i \subset T_i$.

$\phi$ : $\forall Pl_i \in Pl,\ \forall t \in T_i, \quad \phi(t) = \phi_i(t)$ ;

$M_0$ : $\forall (Pl_i, Pl_j) \in Pl \times Pl \ (i \neq j)$
si $M_{0,i}(p) \in M_{0,c}(p)$,
$M_0(p) = M_{0,c}(p)$;
sinon $M_0(p) = M_{0,i}(p)$.

$\Pi$ : $\forall Pl_i \in Pl,\ \forall t \in T_i, \quad \Pi(p) = \Pi_i(p).$

#### Génération d'algorithme

Nous présentons ici l'algorithme générant le modèle global de SGC à partir de ses modes de fonctionnement.

### 6.3.2 Exemple illustratif

#### Description du système

L'exemple retenu dans cette section est SGC composée de trois niveaux de plans :

— $Plan1$ concerne les crises limitées qui n'affectent pas sérieusement la capacité fonctionnelle de la zone concernée et qui ne dépassent pas les ressources disponibles dans cette zone, mais nécessitent néanmoins un certain degré d'action ;

— $Plan2$ concerne les urgences ou les sinistres pouvant être graves et causer des dommages, des pertes de vies humaines ou des blessures et/ou interrompre les opérations de la zone concernée. Ces incidents peuvent découler d'incidents commençant au $Plan1$. Les ressources de la zone concernée et celles des zones voisines sont mises à la disposition ;

— $Plan3$ est destiné aux crises majeures telles que les catastrophes naturelles et peut être provoqué par des incidents commençant au $Plan2$. Dans ce plan, les ressources de la zone concernée, de tous ses voisins et des voisins des voisins sont mises à la disposition.

Comme séries de crise, soit deux explosions coordonnées dans deux trains de voyageurs qui ont fait de très nombreuses victimes. L'explosion du premier train s'est produit dans la zone $A1$. Le $Plan2$ est immédiatement déclenché dans cette zone. Quelques minutes plus tard, le deuxième explosion s'est produit dans la zone $A7$. Le $Plan2$ est également déclenché dans la nouvelle zone.
La Figure 6.3 montre la carte de situation de l'exemple où la couleur jaune représente les zones concernées par les deux $Plan2$ déclenchés. Comme le montre la carte de situation, un chevauchement de zones des deux plans existe. Par conséquent, un nouveau plan est créé résultant de la fusion des deux plans précédents.

#### Modèles RdPCP

Pour mieux comprendre le modèle global, nous décrivons tout d'abord les modèles abstraits des plans correspondant aux différents plans de crise dans des modèles RdPCP distincts. Cette description permet de comprendre tout mode de fonctionnement déclenché dans n'importe quel niveau de plan.

**Algorithme 10 :** Génération du modèle de SGC

**Entrées :** l'ensemble $A$ des zones ;
l'ensemble $Pl$ des plans de crise ;
l'ensemble des modèles RdPCP abstrait des plans de crise ;
le mappage $Next\_plan$ ;
**Output :** Le modèle RdPCP global de SGC $< P, T, K, D, W^-, W^+, \Phi, M_0, \Pi >$

$P \leftarrow \varnothing$;
$T \leftarrow \varnothing$;
$K \leftarrow \varnothing$;
**pour chaque** *plan de crise* $Pl_i, i \leftarrow 1$**à** $|Pl|$ **faire**
  $P \leftarrow P \cup P_i$;
  $T \leftarrow T \cup T_i$;
  $K \leftarrow K \cup K_i$;
  **pour chaque** $(p,t) \in P_i \times T_i$ **faire**
    $W^-(p,t) \leftarrow W_i^-(p,t)$;
    $W^+(p,t) \leftarrow W_i^+(p,t)$;
  **fin**
  **pour chaque** $t \in T_i'$ **faire**
    $W^+(Activated\ Plans, t) \leftarrow Next\_plan(t)$;
  **fin**
  **pour chaque** $t \in T_i$ **faire**
    $\phi(t) \leftarrow \phi_i(t)$;
  **fin**
  **pour chaque** $P \in P_i$ **faire**
    $M_0(p) \leftarrow M_0 \cup M_{0,i}(p)$;
  **fin**
  **pour chaque** $T \in T_i$ **faire**
    $\Pi(t) \leftarrow \Pi_i(t)$;
  **fin**
  **pour chaque** *plan de crise* $PL_j, j \leftarrow 1$**à** $i-1$ **faire**
    **pour chaque** $(p,t) \in P_i \times T_j$, $p \notin P_j$,$t \notin T_i$ **faire**
      $W_i^-(p,t) \leftarrow 0$;
      $W_i^+(p,t) \leftarrow 0$;
    **fin**
  **fin**
**fin**

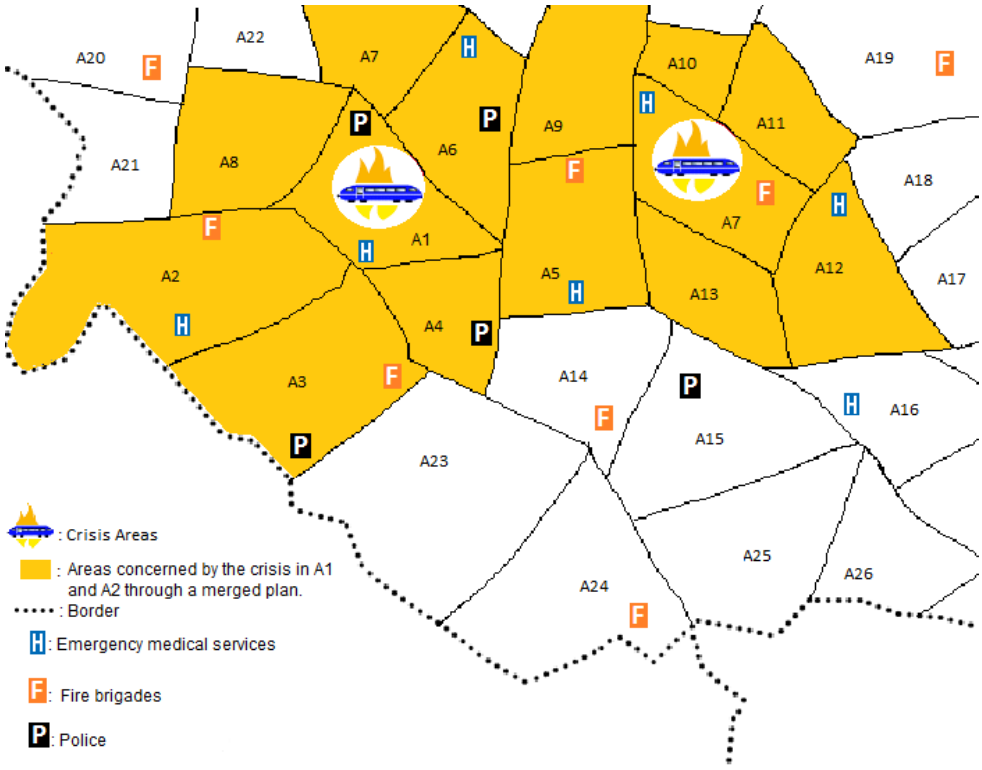



Figure 6.3 – Carte de situation.

Dans l'exemple étudié, il existe quatre modes de fonctionnement de plan : $\{Plan1, Plan2, Plan3, Merged\ Plans\}$. $Merged\ Plans$ contient l'ensemble des plans de crise fusionnés. La Figure 6.4 représente le comportement du SGC dans $Plan1$.

- Le déclenchement/arrêt de $Plan1$ est représenté par la machine à états composée des places $\{Activated\ Plans, Enabled\ Plans, Areas\ Plan1\}$, des transitions $\{Activate\ Plan1, Disable\ Plan\}$ et les arcs qui les interconnectent.
- Chaque jeton de $\{Activated\ Plans, Enabled\ Plans\}$ est constitué de deux listes. La première liste contient le couple de valeurs : la zone de crise et le plan activé. La deuxième liste contient les zones concernées par le plan activé.
- La réservation/libération des ressources autorisées est représenté par les places $\{Idle\ Resources, Active\ Resources\}$ et leurs arcs associés.
- La transition $Change\ Plan1\ to\ Plan2$ est une transitions de commutation et $Merging\ Crises\ Areas$ est une transition prioritaire, elle assure la fusion des plans. Nous avons alors $Next\_plan(Change\ Plan1\ to\ Plan2) = Plan2$ et $Next\_plan(Merging\ Crises\ Areas) = Merged\ Plans$).

La Figure 6.5 (resp. la Figure 6.6) décrit le comportement du fonctionnement dans $Plan2$ (resp. $Plan3$). Nous pouvons noter les points suivants :

- L'activation d'une crise dans le $Plan2$ (resp. $Plan3$) est représentée par la transition $Activate\ Plan2$ (resp. $Activate\ Plan3$). La place $Areas\ Plan2$ (resp.$Areas\ Plan3$)

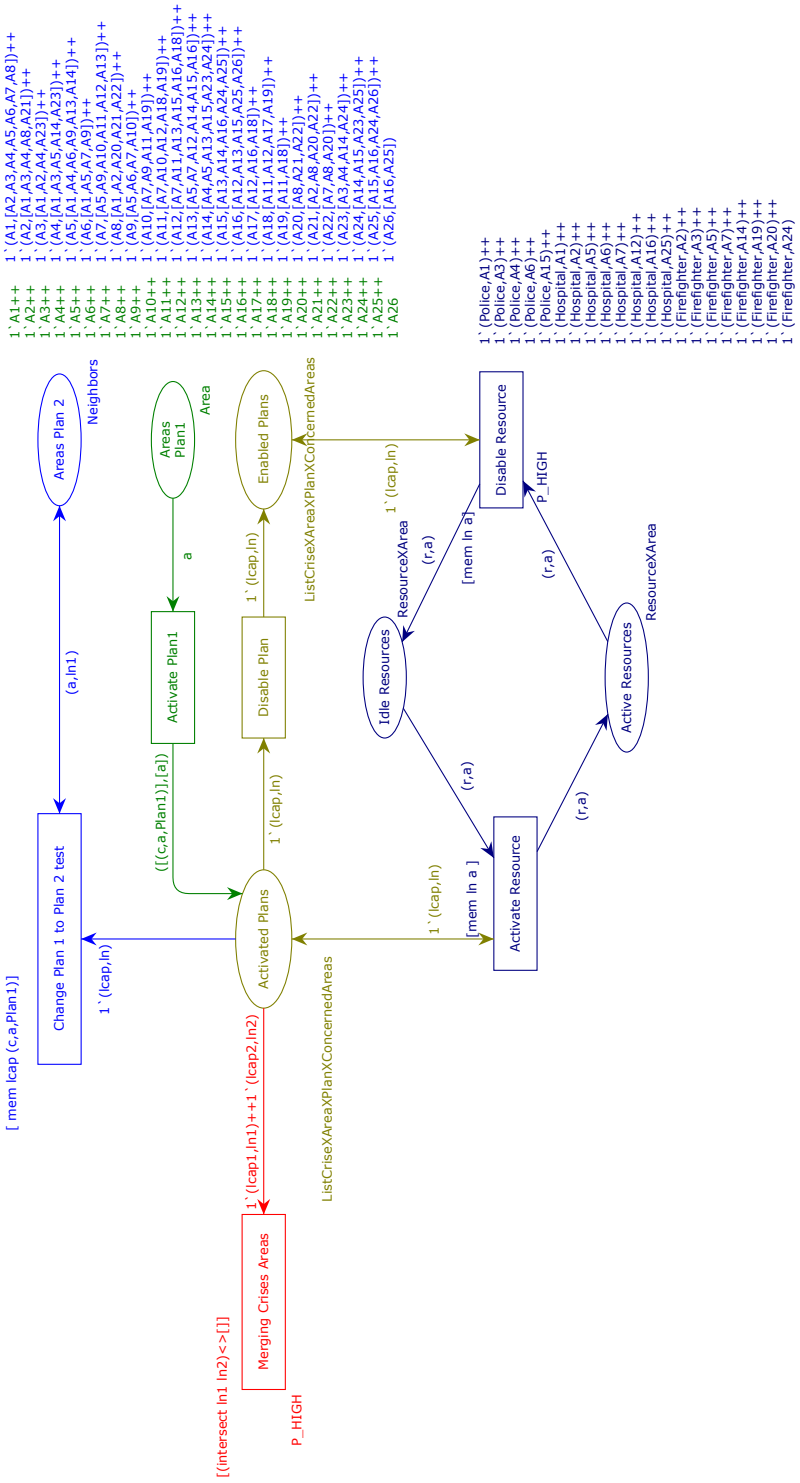



Figure 6.4 – Modèle de $Plan1$

représente la zone de crise, ses zones voisines concernées par la crise lorsque $Plan2$ (resp. $Plan3$) est appliqué.

— Un sous-modèle commun entre les modèles apparaît à travers les places : {$Idle\ Resources$, $Active\ Resources$, $Activated\ Plans$, $Enabled\ Plans$} et transitions {$Activate\ Resource$,

*Disable Resource*, *Disable Plan*, *Merging Crises Areas*}. Les arcs internes sont aussi les mêmes par définition.

— La transition *Change Plan*2 *to Plan*3 est une transition qui permet au système de changer le *Plan*2 en *Plan*3 pour un plan de crise activé en *Plan*2 i.e. *Next_plan* (*Change Plan*2 *to Plan*3) = *Plan*3.

Le fonctionnement du mode *Merged plans*, modélisé à la Figure 6.7, est similaire aux autres plans avec la possibilité de changer le type des plans de crise qui le composent par un autre plan de couverture de zones plus étendues.

La Figure 6.8 présente le RdPCP global décrivant la gestion des modes de fonctionnement du SGC étudié. Dans ce modèle, tous les places, transitions et arcs des quatre modes de fonctionnement sont présents sans dupliquer des sous-modèles communs.

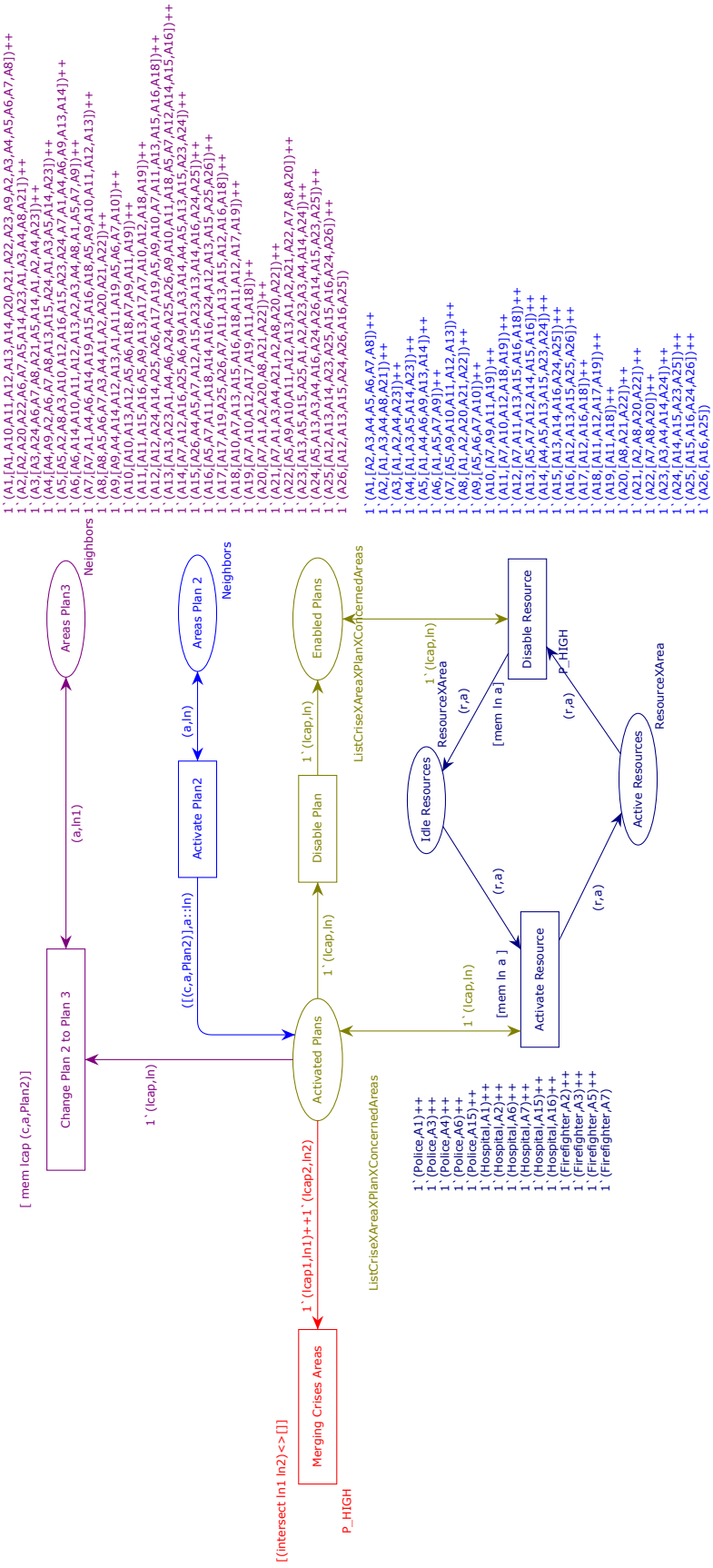



Figure 6.5 – Modèle de *Plan*2

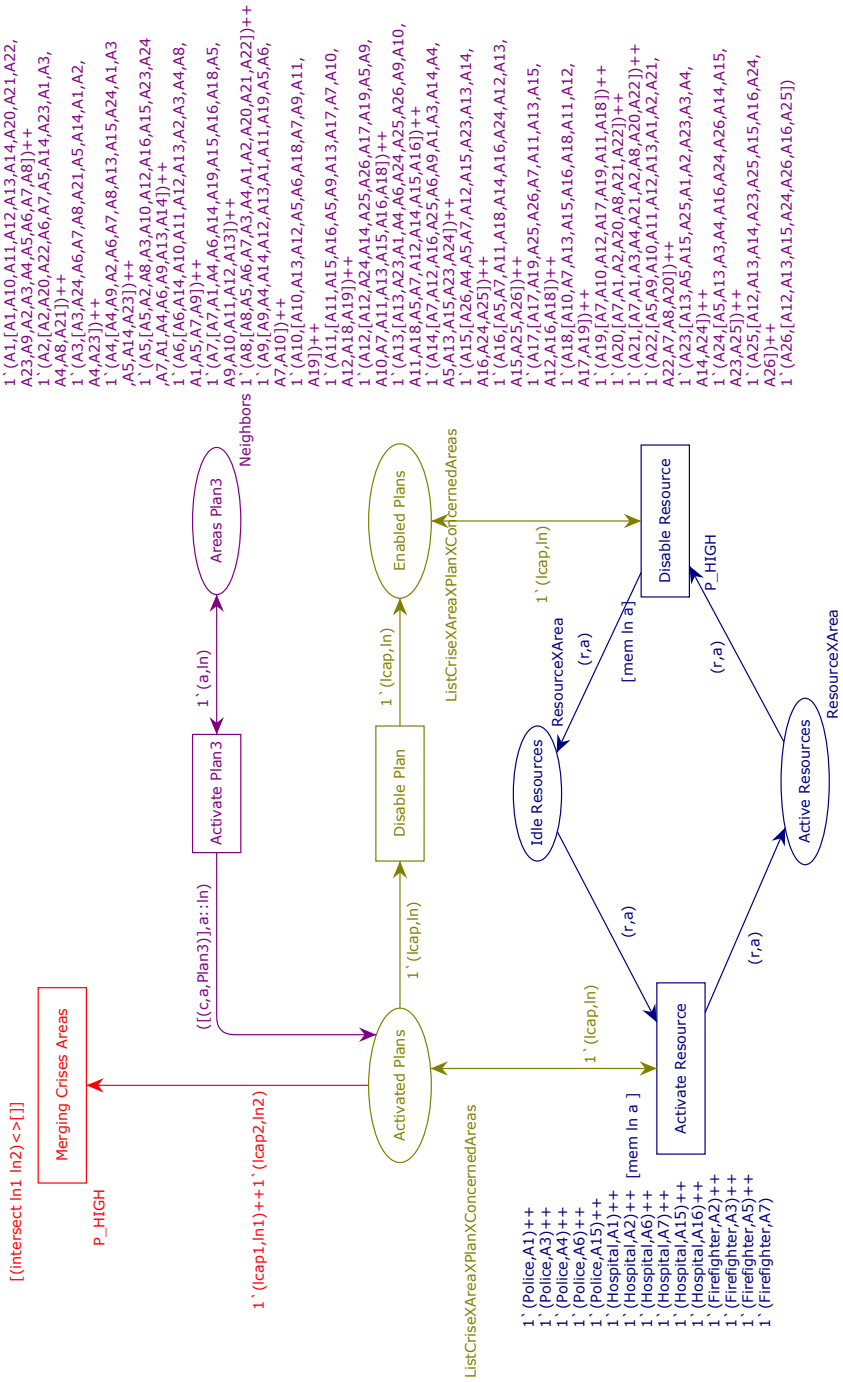



Figure 6.6 – Modèle de *Plan*3

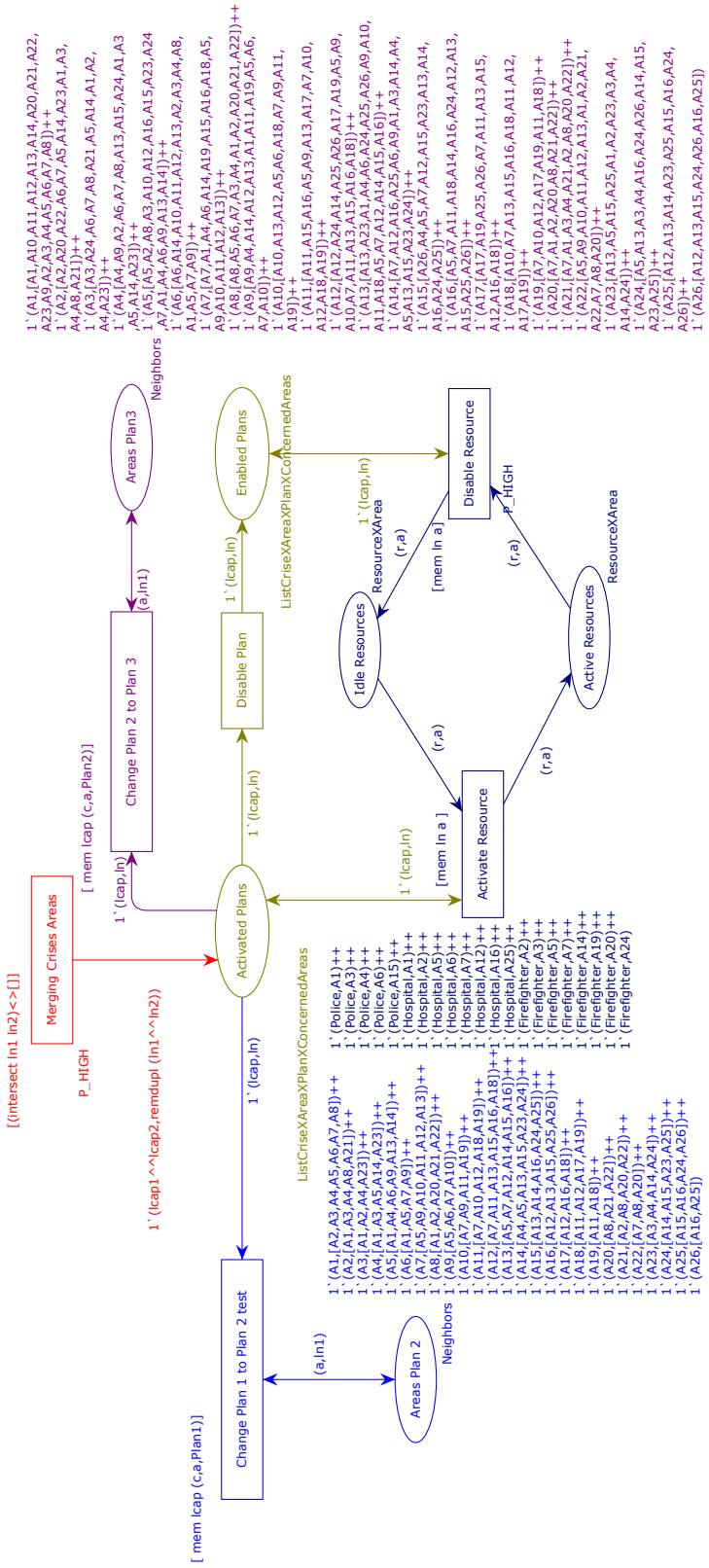



Figure 6.7 – Modèle des plans fusionnés.

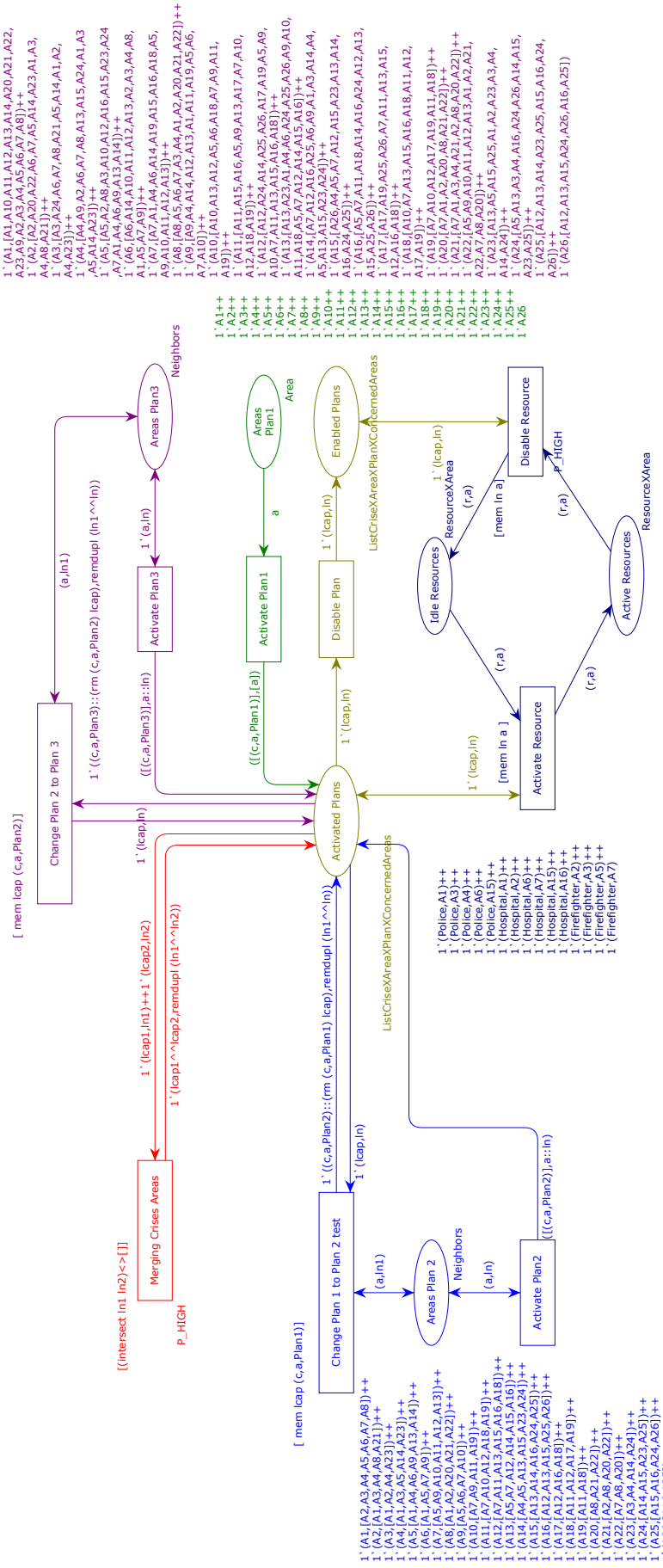



Figure 6.8 – Modèle du SGC

## 6.4 Coopération internationale en gestion de crise

Parce que les frontières politiques et administratives ne constituent pas des obstacles à la propagation des catastrophes naturelles et humanitaires, il doit exister une coopération dans l'UE entre les État membres et, de manière générale entre les pays du monde. Les bonnes pratiques suggèrent que les Systèmes de Gestion de Crise de Pays (SGCPs) coopèrent avec leurs homologues pour une réaction rapide, adaptée et répartie particulièrement pour les crises situées dans les régions frontalières pouvant entraîner des pertes de vies humaines ou des blessures corporelles et/ou des dégâts importants.

Dans cette section, nous proposons une plateforme formelle de coopération internationale entre les SGCPs pour les crises frontalières. La modélisation de cette coopération a abouti au concept de SdS représentant un système international de gestion de crise (SIGC). La structure hiérarchique, illustrée à la Figure 6.9, décrit graphiquement la composition du SdS étudié. Un SIGC se compose de plusieurs SGCP dont chacun comprend plusieurs plans de crise.

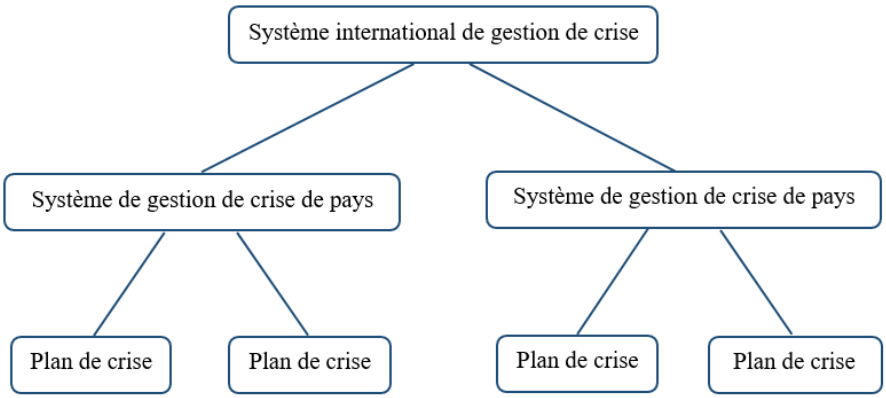



Figure 6.9 – Structure hiérarchique proposée d'un SdS.

### 6.4.1 Modélisation du SIGC

Particulièrement dans les régions frontalières, la gestion de crise est face à plusieurs défis. Une préparation approfondie et une anticipation sont des outils organisationnels cruciaux pour faire face aux engeux et donc permettre une réaction appropriée aux crises. La Figure 6.10 représente un diagramme de classe décrivant la structure du SdS étudié. Notons d'abord que nous confondons le SGCP et son pays. Un pays est composé de plusieurs zones et possède plusieurs plans de crise. Une crise peut déclencher plusieurs plans et toucher plusieurs zones. Chaque zone dispose de plusieurs ressources : policiers,

pompiers, services médicaux d'urgence, etc. Le contrôle des ressources disponibles et la coordination sont déjà déterminés par le biais d'un processus de prise de décision clair.

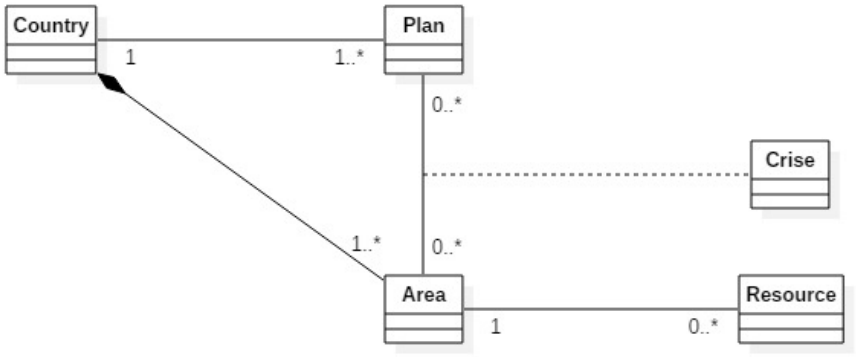



Figure 6.10 – Diagramme de classe du SIGC

Cette section se base sur les mêmes concepts et approches que la section précédente pour la modélisation des SGCPs à savoir la gestion des modes de fonctionnement et l'approche multi-modèle. Ainsi, nous retrouvons les mêmes définitions avec l'ajout de l'indice du pays.

**Définition 40 (Définitions lié au SIGC)**
***L'ensemble des systèmes de gestion de crise nationaux*** *est appelé* $Cn = \{cn_1, cn_2, .., cn_{|Cn|}\}$ *où* $|Cn| > 1$.
$\forall cn_i \in Cn$,

***L'ensemble des plans de crises*** *de* $cn_i$ *est appelé* $Pl_i = \{pl_{i,1}, pl_{i,2}, .., pl_{i,|Pl_i|}\}$ *où* $|Pl_i| \geq 1$.

***L'ensemble des zones*** *de* $cn_i$ *est appelé* $A_i = \{a_{i,1}, a_{i,2}, ..., a_{i,|A_i|}\}$ *où* $|A_i| \geq 1$.

***L'ensemble des plans de crise activés*** *de* $cn_i$ *est appelé* $PL_i$ *où*

$$\begin{array}{ccc} A_i \times Pl_i & \longrightarrow & PL_i \\ (a_{i,j}, pl_{i,k}) & \longmapsto & PL_{i,j,k} \end{array}$$

*est une application tel que* $PL_{i,j,k}$ *indique le plan* $pl_{i,k}$, $k = 1..|Pl_i|$, *activé dans la zone de crise* $a_{i,j}$, $j = 1..|A_i|$.

***L'ensemble des zones de crises*** *concernés par* $PL_{i,j,k}$ *est appelé* $A_{i,j,k}$. *où* $i \in 1..|Cn|$, $j \in 1..|A_i|$ *et* $k \in 1..|Pl_i|$.

De même que dans la section précédente, à chaque mode de fonctionnement $PL_{i,j,k}$,$i \in 1..|Cn|$, $j \in 1..|A_i|$ et $k \in 1..|Pl_i|$, nous associons un modèle RdPC $< P_{i,j,k}, T_{i,j,k}, K_{i,j,k}, D_{i,j,k}, W^-_{i,j,k}, W^+_{i,j,k}, \phi_{i,j,k}, M_{0,i,j,k} >$ et pour des raisons de simplicité, la même identité de

chaque mode du $PL_{i,j,k}$ est attribuée à son RdPCP associé.
Le concept de Modèle abstrait de plan de crise et de sous-modèle commun et le mécanisme de commutation sont aussi appliqués dans cette section (pour plus de détail, voir les sous-sections 6.3.1).

**Définition 41 (Modèle abstrait de plan de crise)**
*Un mode de fonctionnement* $PL_{i,k} =< P_{i,k}, T_{i,k}, K_{i,k}, D_{i,k}, W^-_{i,k}, W^+_{i,k}, \phi_{i,k}, M_{0,i,k} >$, $i \in 1..|Cn|$ *et* $k \in 1..|Pl_i|$, *est dit modèle abstrait de plan de crise si, et seulement si,* $\forall < P_{i,j,k}, T_{i,j,k}, K_{i,j,k}, D_{i,j,k}, W^-_{i,j,k}, W^+_{i,j,k}, \phi_{i,j,k}, M_{0,i,j,k} >$ *où* $j \in 1..|A_i|$,

$$< P_{i,k}, T_{i,k}, K_{i,k}, D_{i,k}, W^-_{i,k}, W^+_{i,k}, \phi_{i,k} >=$$

$$< P_{i,j,k}, T_{i,j,k}, K_{i,j,k}, D_{i,j,k}, W^-_{i,j,k}, W^+_{i,j,k}, \phi_{i,j,k} > .$$

$$et\ M_{0,i,j,k} \subseteq M_{0,i,k}.$$

**Définition 42 (Sous-comportement commun)**
*Soient* $cn_i \in Cn$ *et Soit* $PL_i$ *son ensemble des modes de fonctionnement.*
$\forall (PL_{i,j,k}, PL_{i,l,m}) \in PL_i \times PL_i$ *où* $j \neq l$ *et* $m \neq m$,
*si* $\exists c =< P_c, T_c, K_c, D_c, W^-_c, W^+_c, \phi_c, M_{0,c} >$ *tels que* $(P_c = (P_{i,j,k} \cap P_{i,l,m}) \neq \emptyset)$ *ou* $(T_c = (T_{i,j,k} \cap T_{i,l,m}) \neq \emptyset)$*alors c est un sous-comportement commun aux deux modes* $PL_{i,j,k}$ *et* $PL_{i,l,m}$ *si et seulement si*

*(a)* $\forall (p,t) \in P_c \times T_c, \quad W^-_{i,j,k} = W^-_{i,l,m}$ *et* $W^+_{i,j,k} = W^+_{i,l,m}$,

*(b)* $\forall t \in T_c, \quad \phi_{i,j,k} = \phi_{i,l,m}$.

**Définition 43 (Mécanisme de commutation)**
*Soit* $PL_{i,j,k}$, $i \in 1..|Cn|$, $j \in 1..|A_i|$ *et* $k \in 1..|Pl_i|$ *un mode de fonctionnement et* $T'_{i,j,k} \subset T_{i,j,k}$ *un ensemble de transitions de commutation.*
*Soit* $Next\_plan : T'_{i,j,k} \rightarrow PL_i$ *une application tel que* $Next\_plan(t)$ *indique le mode de fonctionnement actif après le déclenchement de* $t$, $\forall t \in T'_{i,j,k}$.

### Modèle de système de gestion de crise nationaux

Sur la base des définitions précédentes, nous sommes en mesure de modéliser les systèmes de gestion de crise nationaux. D'abord, soit, pour chaque SGCP, la place contenant les jetons représentants les plans de crise activés est appelée *Activated Plans Country* $i$, $i \in 1..Cn$.
Pour chaque pays $cn_i \in Cn$, le RdPC associé est un tuple

$$< P_i, T_i, K_i, D_i, W^-_i, W^+_i, \phi_i, M_{i,0} >$$

tel que

$P_i = \cup_{(j=1..|PL_i|,k=1..|A_i|)} P_{i,j,k}$;

$T_i = \cup_{(j=1..|PL_i|,k=1..|A_i|)} T_{i,j,k}$;

$K_i = \cup_{(j=1..|PL_i|,k=1..|A_i|)} K_{i,j,k}$;

$D_i$ est défini, par extension, de $P_i \cup T_i$ dans l'ensemble des domaines de couleur ;

$W_i^-, W_i^+$ :

— $\forall PL_{i,j,k} \in PL_i, \forall (p,t) \in P_{i,j,k} \times T_{i,j,k}$,

$$W_i^-(p,t) = W_{i,j,k}^-(p,t),$$

$$W_i^+(p,t) = W_{i,j,k}^+(p,t);$$

— $\forall (PL_{i,j,k}, PL_{i,l,m}) \in PL_i \times PL_i \ (k \neq m),\ \forall (p,t) \in P_{i,j,k} \times T_{i,l,m},\ p \notin P_{i,l,m},\ t \notin T_{i,j,k}$,

$$W_i^-(p,t) = W_i^+(p,t) = 0;$$

— $\forall PL_{i,j,k} \in PL_i, \forall t \in T'_{i,j,k}$,

$$W^+(Activated\ Plans\ Country\ i, t) = Next\ plan(t);$$

$\phi_i$ : $\forall PL_{i,j,k} \in PL_i$ et $t \in T_{i,j,k}$,

$$\phi_i(t) = \phi_{i,j,k}(t);$$

$M_{0,i}$ : $\forall PL_{i,j,k} \in PL_i$ et $\forall p \in P_{i,j,k}$,

$$M_{0,i} = M_{0,i,j,k}.$$

**Coopération inter-systèmes**

La coopération inter-SGCPs en cas de crise fournit de précieuses ressources supplémentaires au pays victime. Dans notre proposition, en cas de coopération, le système voisin déclenche un plan de crise dans la zone frontalière voisine de la zone de crise initiale afin de mettre ses ressources, ou plutôt ses services, à la disposition du pays en crise.
La coopération est assurée par un mécanisme composé principalement d'un nouvel ensemble de transitions, dont le rôle est de démarrer et terminer la coopération. On ne peut pas prévoir un mécanisme spécifique de coopération dans le cadre de gestion de crise. Pour cela, nous le notons $S_c$ et qu'il est modélisé par un RdPC $< P_c, T_c, K_c, D_c, W_c^-, W_c^+, \phi_c, M_{c,0} >$.

**Modèle SdS**

Nous présentons ici le modèle global du SIGC à partir des modèles correspondant à ses systèmes et de leurs mécanismes de commutation.

Un SIGC est un tuple $< P, T, K, D, W_-, W_+, \phi, M_0 >$ où

$P = \bigcup_{(i=1..|Cn|)} P_i \cup P_c$;

$T = \cup_{(i=1..|Cn|)} T_i \cup T_c$;

$K = \cup_{(i=1..|Cn|)} K_i \cup K_c$;

$D$ est défini, par extension, de $P \cup T$ dans l'ensemble des domaines de couleur ;

$W^-, W^+$ :

— $\forall cn_i \in Cn$, $\forall (p,t) \in P_i \times T_i$,

$$W^-(p,t) = W_i^-(p,t) \text{ et } W^+(p,t) = W_i^+(p,t),$$

— $\forall (cn_i, cn_j) \in Cn \times Cn (i \neq j)$, $\forall (p,t) P_i \times T_j$, $p \notin P_j$ et $t \notin T_i$,

$$W^-(p,t) = W^+(p,t) = 0;$$

— $\forall (P,t) \in P_c \times T_c$,

$$W^-(p,t) = W_c^-(p,t) \text{ et } W^+(p,t) = W_c^+(p,t);$$

$\phi$ : $\forall cn_i \in Cn, \forall t \in T_i, \phi(t) = \phi_i(t)$;
$\forall t \in T_c$, $\phi(t) = \phi_c(t)$;

$M_0$ : $\forall cn_i \in Cn, \forall p \in P_i$, $M_0(p) = \cup M_{0,i}(p)$;
$\forall P \in P_c$, $M_0(p) = M_{0,c}(p)$.

### 6.4.2 Exemple illustratif

#### Description du système

L'objectif de cette sous-section est d'illustrer l'application concrète de l'approche proposée sur un exemple d'accident ferroviaire. L'exemple retenu est une explosion dans un train de voyageurs de $Country1$ qui a fait de très nombreuses victimes près de la frontière de $Country2$, comme le montre le carte de situation de la Figure 6.11. Un plan de crise est immédiatement déclenché pour en limiter au maximum les impacts. Comme la priorité est de sauver des vies et d'empêcher des pertes humaines supplémentaires, $Country1$ demande l'aide de $Country2$ dans le but de réagir encore plus rapidement et de tirer parti des ressources et services potentiellement disponibles. Nous avons alors $Cn = Country1, Country2$.

Du côté organisation, chaque pays est subdivisé en zone permettant de préciser la (ou les) zone(s) sinistrée(s) ainsi que les zones couvertes par le plan de crise activé. Les

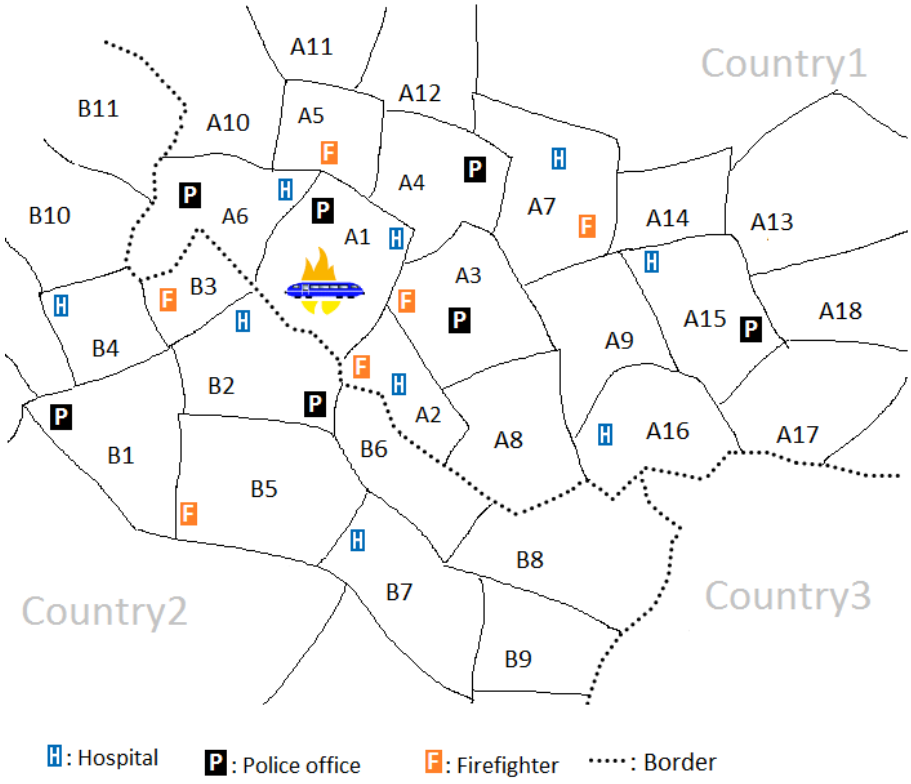



Figure 6.11 – Carte de situation.

zones de couverture de crise sont des zones qui mettent à disposition du décideur leurs ressources : personnel préfectoral, policiers, gendarmes, pompiers, services médicaux d'urgence et même militaires. Le décideur est la personne responsable des échanges avec les pouvoirs publics et de déclenchement du plan de crise.
L'ensemble des zones de $Country1$ est appelé $A_1$ et celui de $Country2$ est $A_2$.
$A_1 = \{A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9, A_{10}, A_{11}, A_{12}, A_{13}, A_{14}, A_{15}, A_{16}, A_{17}, A_{18}\}$ et $A_2 = \{B_1, B_2, B_3, B_4, B_5, B_6, B_7, B_8, B_9, B_{10}, B_{11}\}$.
$Country1$ possède trois plans de crise de complexité variable $P_1 = Plan1, Plan2, Plan3$ pendant que $Country2$ ne possède que deux : $P_2 = PlanYellow, PlanRed$.
Le $Plan1$ concerne une crise limitée qui n'affectera pas sérieusement la capacité fonctionnelle de la zone sinistrée, mais qui nécessite néanmoins un certain degré d'action et seules les ressources de la zone concernée peuvent être disponibles.
$Plan2$ concerne les urgences ou les sinistres graves pouvant causer des dommages, des pertes de vie ou des blessures et/ou interrompre les opérations de la zone concernée et pouvant découler d'incidents commençant au $Plan1$. Les ressources de la zone concernée et celles des zones voisines sont mises à la disposition du décideur. Le $Plan3$ concerne les crises majeures, telles que les catastrophes naturelles, qui peut être provoquées par des incidents commençant au $Plan2$. Dans ce plan, les ressources d'une couverture de zone

plus importante sont mises à disposition. Les plans $PlanYellow$ et $PlanRed$ du $Country2$ sont similaires aux plans $Plan1$ et $Plan2$ respectivement.
Puisque sauver et préserver des vies est un objectif premier, $Country1$ demande l'aide de $Country2$ dans le but de réagir le plus rapidement possible à cette situation d'urgence et de tirer parti des ressources étroites existantes situées au-delà de la frontière. $Country2$ déclenche logiquement une crise dans une zone frontalière proche de la crise pour mettre ses ressources au disposition du décideur du $Country1$ et selon le besoin, $Country2$ peut changer le plan activé.

### Modèles RdPC

La Figure 6.19 représente le modèle RdPC de l'ensemble du SdS. Les Figures 6.15 et 6.18 montrent les deux SGCPs constituants le SIGC. Elles décrivent le fonctionnement internes de chaque système en présence de crise. Les Figures 6.12,6.13,6.14,6.16 et 6.17 représentent respectivement les modèles RdPC de $Plan1$, $Plan2$ et $Plan3$ de $Country1$ et $PlanYellow$ et $PlanRed$ de $Country2$.
La description et la construction des modèles des SGCP sont déjà détaillées dans la section précédente.
Comme le montre la Figure 6.19, le SIGC se compose du SGC de $Country1$ et du système du SGC de $Country2$ qui fonctionnent en parallèle. La Figure 6.12 représente le comportement du SGC dans $Plan1$.

— Le déclenchement/arrêt de $Plan1$ est représenté par les places $\{Activated\ Plans\ Country1, Areas\ Plan1\}$, des transitions $\{Activate\ Plan1, Disable\ Plan1\}$ et les arcs qui les interconnectent.

— La réservation/libération des ressources est représenté par les places $\{Idle\ Resources\ Country1, Active\ Resources\ Country1\}$, les transitions $\{Activate\ Resource\ Country1$, $Disable\ Resource\ Country1$ et leurs arcs associés. Cette partie représente un sous-comportement commun avec les les autres modes du même système.

— La transition $Change\ Plan1\ to\ Plan2$ est une transition de commutation. Nous avons alors $Next\_plan(Change\ Plan1\ to\ Plan2) = Plan2$.

Les modèles de $Plan2$ et $Plan3$ du $Country1$ et de $PlanYellow$ et $PlanRed$ du $Country2$ ressemblent à celui du modèle de $Plan1$ tout en remplaçant le nom du plan et du pays par son correspondant.
Les modèles de SGCPs représentent la gestion des plans en un modèle global où toutes les places, transitions et arcs des modes composants sont présent sans duplication.
Le modèle final du SIGC comprend les deux modèles de SGCPs. Par ailleurs, de nouveaux éléments sont ajoutés pour exprimer la coopération :

— La place *Neighboring Areas of country*1 qui contient la liste des zones frontalières de *country*1 et *Country*2. Plus précisément, pour chaque zone de *Country*1, on lui associe une liste des zones adjacentes de *Country*2.

— La transition *Begin Cooperation Country*1 *with Country*2 dont le rôle est de démarrer la coopération de *Country*2 avec *Country*1. Cette transition active un plan de crise de *Country*2 dans une zone de son territoire adjacente à la zone de crise de *Country*1.

— La transition *End Cooperation Country*1 *with Country*2 qui a pour rôle de mettre fin à la coopération en cours sur demande de *Country*1.

— Les arcs interconnectant dont le rôle est d'assurer le fonctionnement de la place et des transitions décrit ci-dessus.

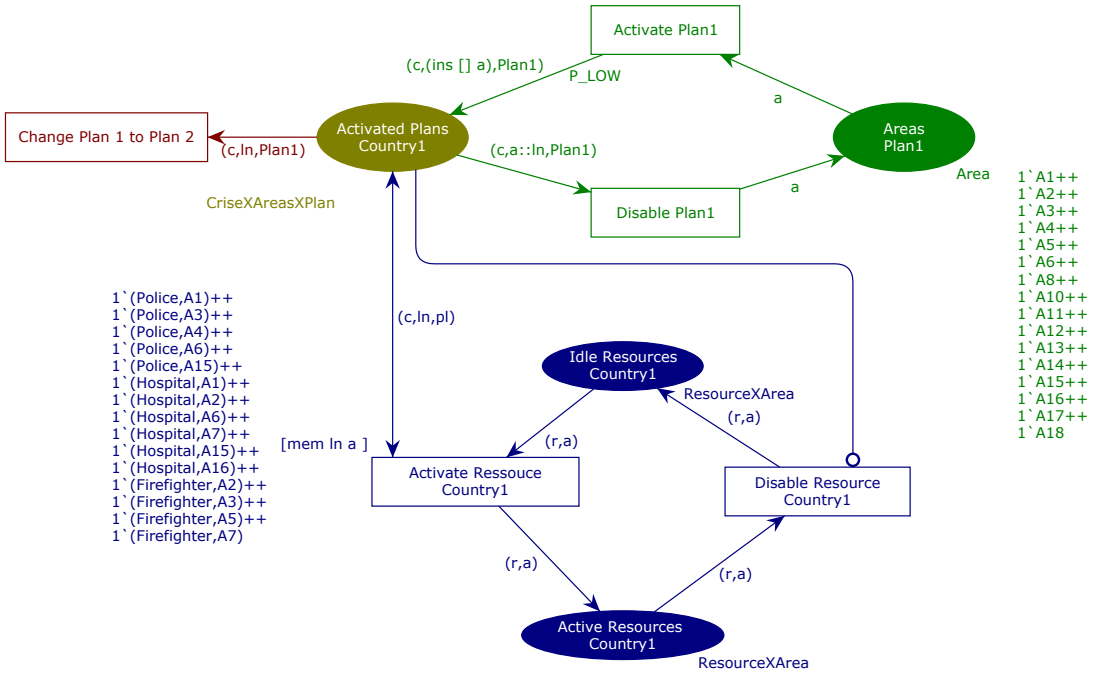



Figure 6.12 – Modèle de *Plan*1

## 6.5 Conclusion

Dans cette section, nous avons argumenté à partir des bonnes pratiques, des normes et de l'état de l'art, que les collaborations centralisées entre différentes parties prenantes ont un impact très positif sur la gestion de crise. Par contre, selon nos investigations, ces mécanismes ne sont pas encore documentés dans les zones frontalières. Les recommandations développées dans ces lignes sont un premier pas pour aborder ce problème, mais elles doivent être étendues et validées dans les travaux futurs.

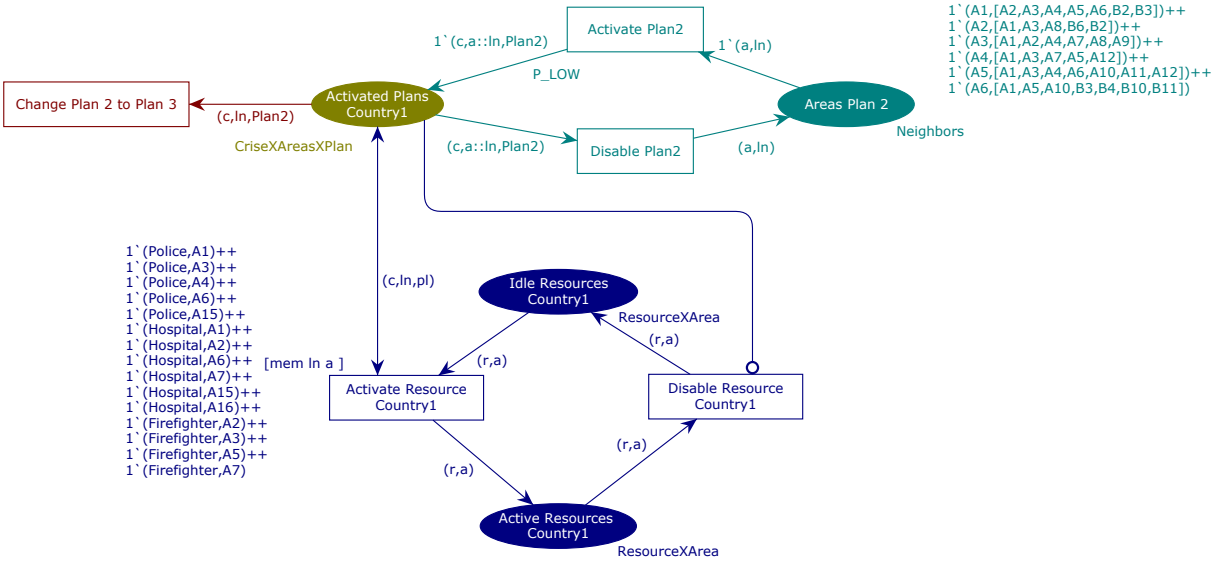



Figure 6.13 – Modèle de *Plan*2

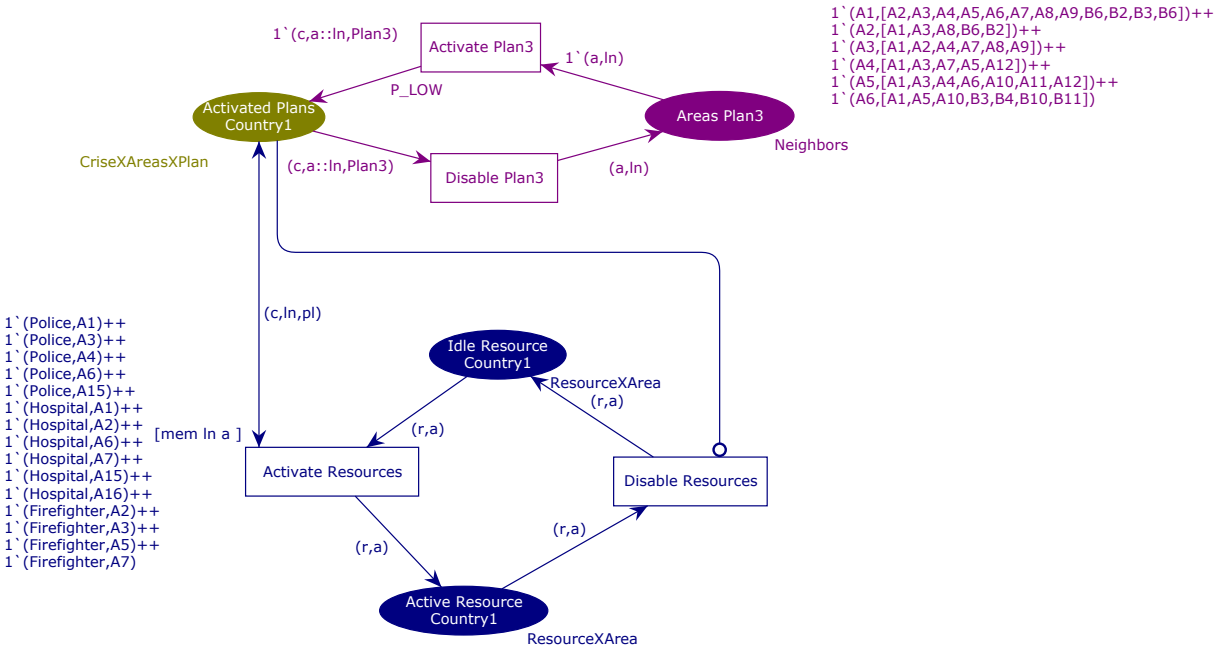



Figure 6.14 – Modèle de *Plan*3

Par ailleurs, nous proposons une méthode de modélisation d'un SdS représentant un système international de gestion des crises. Notre approche prend en compte la dynamique aux seins des systèmes composants et qui influence le fonctionnement global du SdS.
Les SGCs sont approprié pour être considéré comme un SdS car l'approche SdS fournit une vue globale de haut -niveau des systèmes. De plus, la gestion des modes de fonctionnement au sein des systèmes de crise peut être facilement analysée au niveau des SdSs. À cause de l'utilisation des notions de priorité, de simultanéité et de synchronisation, les RdPC se sont avérés appropriés pour modéliser le comportement des SGCs.

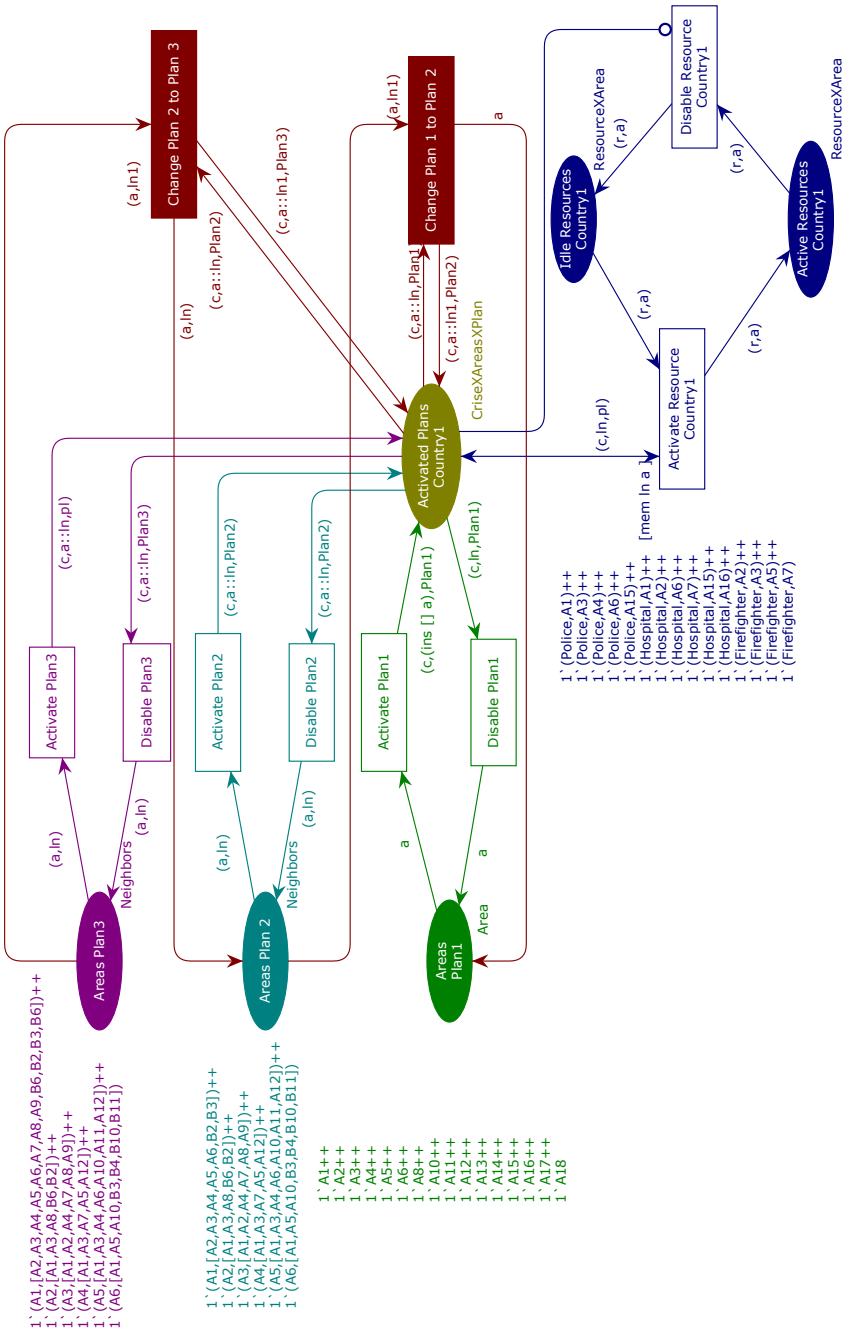



Figure 6.15 – Modèle du SGC de *Country*1.

Nous espérons que cette recherche inspirera d'autres recherches sur le développement d'une plateforme formelle facilitant la coopération en gestion de crise, particulièrement dans les zones frontalières où l'anticipation et la préparation sont essentielles pour pouvoir réagir rapidement et correctement. Pour notre part, les problèmes de gestion de crise dans les zones multinationales sont clairement priorisés pour des travaux futurs. Dans cette situation, les problèmes de cohérence entre les différentes règles deviennent complexes et critiques.

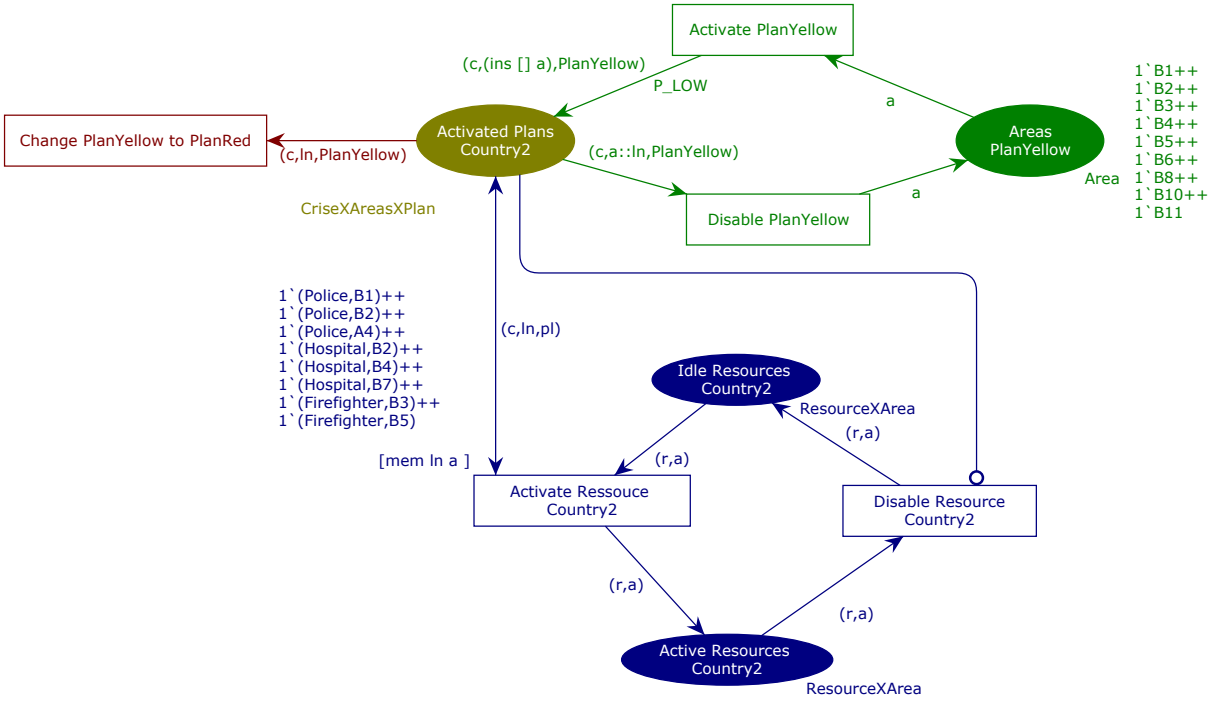



Figure 6.16 – Modèle de *Plan Yellow*

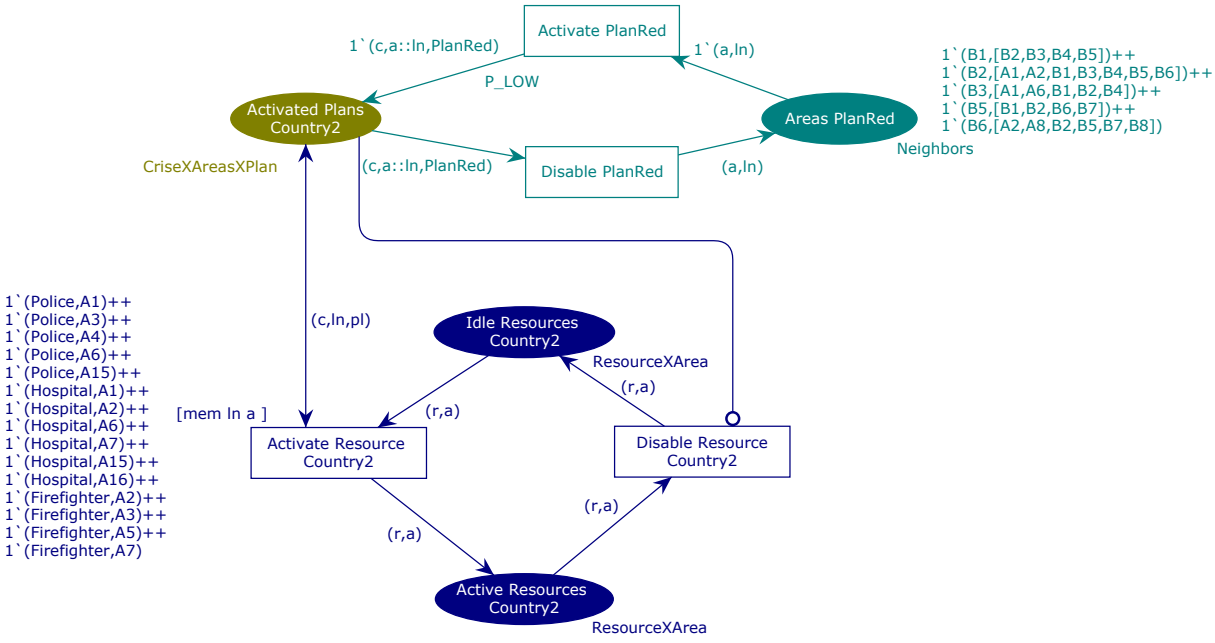



Figure 6.17 – Modèle de *Plan Red*

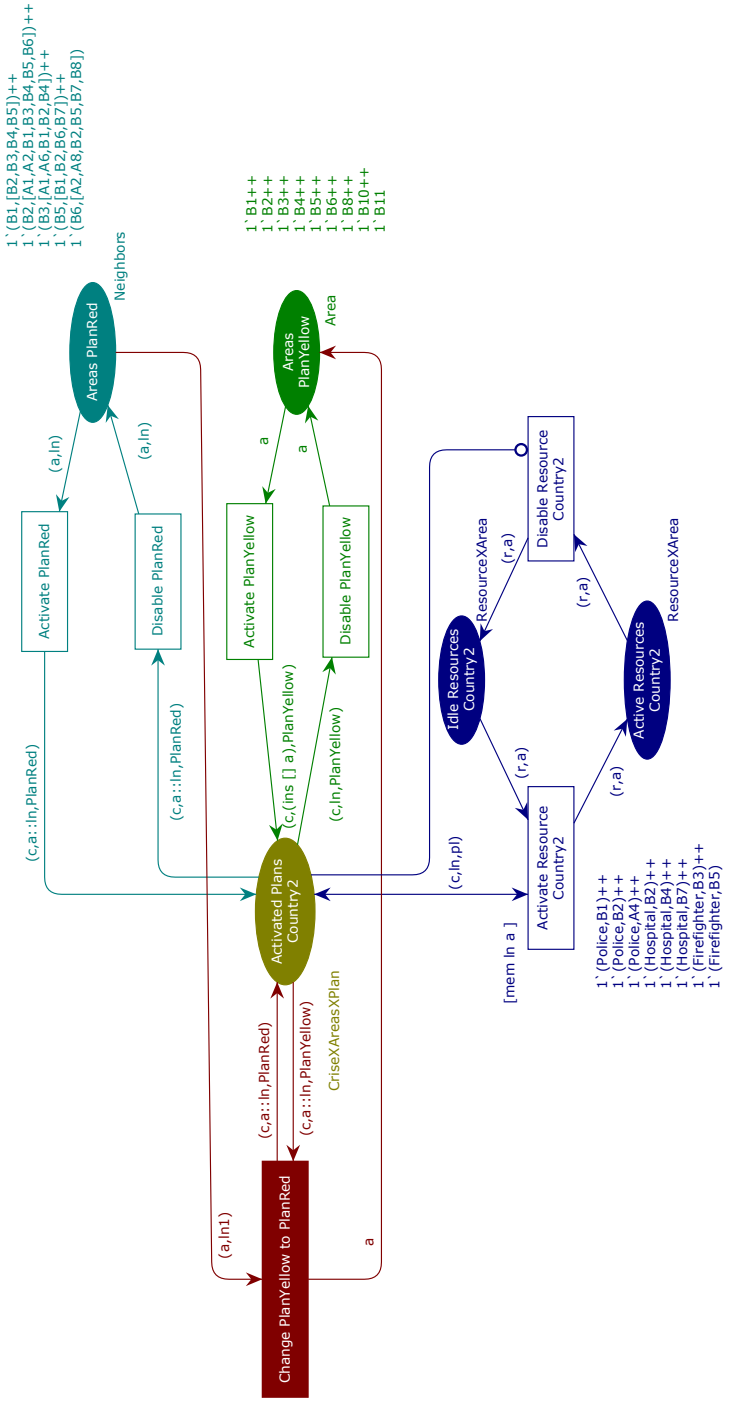



Figure 6.18 – Modèle du SGC de *Country*2.

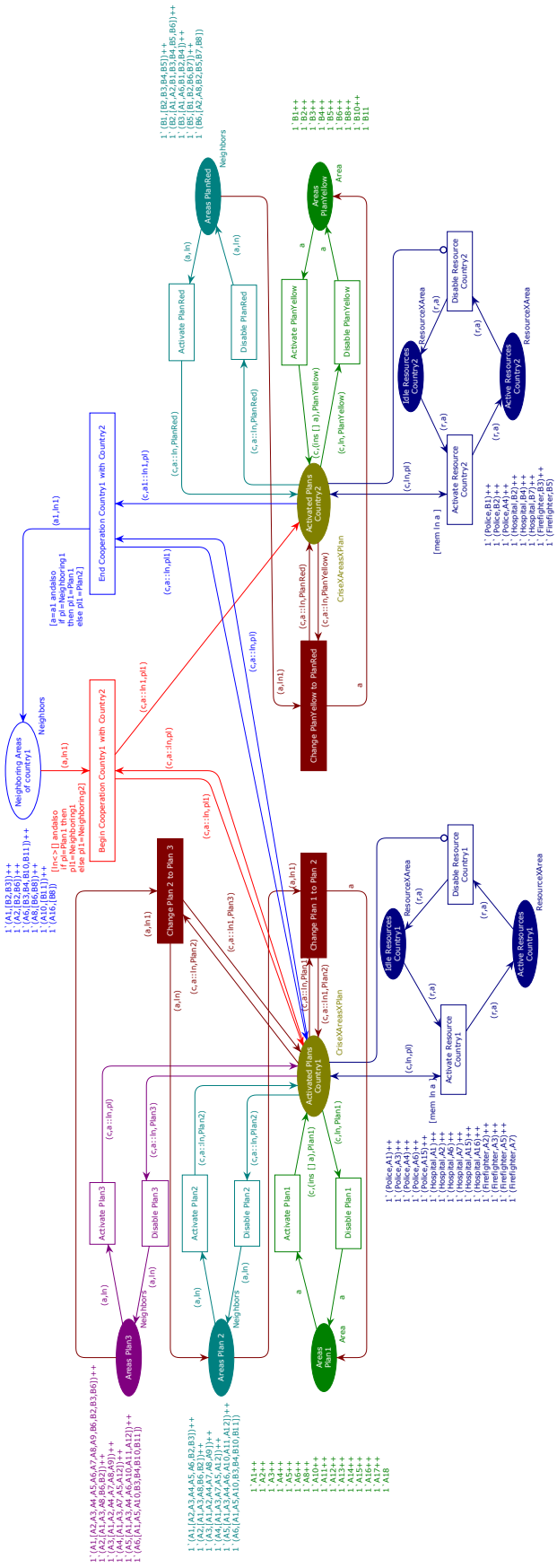



Figure 6.19 – Modèle du SIGC.

# Chapitre 7

# Conclusion

## 7.1 Conclusion

Dans cette thèse, nous avons proposé une démarche de modélisation des SdSs. Le problème du départ était de trouver un moyen sûr pour les trains à grande vitesse de franchir la frontière sans reconfiguration lourde ou arrêt sous ERTMS Niveau 2 dans le cadre du trafic ferroviaire de l'UE. La difficulté dans ce cas, réside dans le fait que le système ERTMS se caractérise par sa complexité et la multiplicité de ses intervenants. En effet, ERTMS Niveau 2 est composée de deux systèmes : le système de bord, installé dans des trains qui appartiennent à plusieurs entreprises ferroviaires, et le système sol, installé dans des infrastructures qui appartiennent à leur tour à des opérateurs ferroviaires nationaux qui fixent chacun leurs propres règles de fonctionnement ERTMS pour se conformer aux directives nationales de sécurité fixées par le gouvernement dont il dépend.

Notre objectif a été alors de disposer d'une vue d'ensemble pour le contrôle des trains d'une part ; et d'autre part, de se doter d'une approche formelle puissante pour la construction de modèles sûrs pour le système ERTMS. Le travail présenté dans ce mémoire concerne une démarche de conception des SdSs vue en tant que SED qui intègre la gestion des modes de fonctionnement au sein de la TCS. Un mode de fonctionnement est un comportement spécifique du système pendant une période de fonctionnement et engageant un ensemble réduit de composants. Nous avons adopté l'approche multi-modèle où chaque mode de fonctionnement est décrit par un modèle RdPHN. La difficulté de la modélisation du SdS est la taille du modèle alors que celle de la gestion des modes de fonctionnement est l'élaboration des modes, leur activation/désactivation, et la possibilité de commuter entre eux.

Dans la première partie, nous avons présenté le contexte industriel dans lequel nous avons décrit en détail le système ERTMS, ses niveaux et ses modes opératoires et nous avons énoncé la problématique scientifique. Ensuite, nous avons fait un état de l'art dans

lequel nous présentons le concept du SdS, la TCS, la gestion des modes, l'approche multi-modèle et les RdPHNs.
Dans la deuxième partie, nous avons proposé une démarche ascendante complètement définie pour la construction d'un modèle du système à l'aide des RdPCs. La première étape est la construction des modèles des composants. La deuxième étape est appelé Modèle des modes où un modèle est construit, indépendamment des autres, pour chaque mode de fonctionnement. La troisième étape, Modèle du système, consiste à la fusion des modèles de modes et à l'intégration du mécanisme de commutation pour l'élaboration d'un modèle système global.

Dans la troisième partie, nous avons présenté une démarche hiérarchique pour la modélisation des SdSs en soulignant les dépendances inter-systèmes qui résulte des reconfigurations intra-système. Le cas de dysfonctionnement de composant est pris en exemple. Les RdPCHs sont utilisés pour modéliser les comportements dynamiques du SdS. Ceci nous a permis de fournir une modélisation et un contrôle de la dynamique intra et inter-systèmes par étapes.
De plus, un ensemble de propriétés a été validé en exploitant le graphe de marquage du modèle obtenu pour vérifier que le comportement du modèle est bien conforme aux attentes.

Dans la quatrième partie, nous avons répondu à notre objectif principal. Le système ERTMS est considéré alors comme un SdS dont le comportement du couple (train, système sol) est vue en tant que mode de fonctionnement. Des modes de fonctionnement transitoires spécifiques ont été ajoutés pour contrôler le comportement des trains dans les zones frontalières. La gestion des modes a permis aux trains d'intégrer le changement de système sol assez naturellement dans le modèle, ce qui a été illustré sur un exemple.
La validation formelle du modèle ERTMS a pu être effectuée pour un certains nombre de propriétés importantes de l'exemple traité.

Dans la dernière partie, nous avons appliqué nos résultats dans le domaine de la gestion de crise. Dans un premier temps, les difficultés de la gestion de crise sont soulignées. Des recommandations pour améliorer la réponse aux situations de crise dans les zones frontalières sont formulées sur la base des normes et directives existantes, des projets de recherche en cours et d'une enquête réalisée auprès des experts en gestion de crise. Dans cette optique, un modèle de gestion de crise est mis en place assurant la fusion et le changement des plans en exploitant les résultats du chapitre 3. Un autre modèle permettant une coopération internationale en cas de crise frontalière est aussi développé en s'appuyant sur les propositions du chapitre 4.

## 7.2 Perspectives

L'extension de la TCS pour la gestion des modes de fonctionnement a été proposée pour la première fois dans les travaux de [Nourelfath 1997]. Depuis, elle a été appliqué dans le cadre des SAP uniquement et à l'exception de [Kadri et al. 2014] qui l'ont appliqué au système ERTMS. Dans cette thèse, nous avons encore changé le cadre d'application pour l'étendre au concept très général des SdS et aux contextes spécifiques du ferroviaire et de la gestion de crise.

Une perspective à long terme pour nos travaux serait la proposition d'outils formels s'appuyant sur l'ingénierie de modèle pour ces deux domaines afin de les rendre plus efficaces d'un point de vue ergonomique pour les experts des domaines concernés. Derrière cet aspect méthodes formelles, se cache une problématique de cohérence sémantique pour laquelle l'ingénierie ontologique pourrait être mobilisée. Des briques ont déjà été réalisées, notamment la transformation de RdPHN en machine abstraites B [Sun 2015, Boudi et al. 2015]. Dans certains cas, un DSL pourrait être aussi utilisé. Enfin, les approches formelles de traçabilité des exigences (l'approche GORE ou CHAOS) devraient permettre de s'assurer que le service demandé est bien celui qui est effectué.

Il faudrait de plus envisager une étude de sécurité-confidentialité de l'architecture proposée, puisqu'on est dans le contexte hautement critique de la gestion de crise. Sur un plan plus large, la résilience de la cellule de gestion de crise devra faire l'objet de toutes les attentions. Nous pourrons, par exemple, mobiliser l'état de l'art sur le recouvrement ou la tolérance aux fautes. Une deuxième perspective à nos travaux serait d'étendre nos résultats à l'orchestration des services critiques pour les transports multimodaux. De fait, des travaux collaboratifs préliminaires avec des partenaires du domaine de l'avionique concernant la gestion de crises des hub de transports multimode ont donné lieu à une publication en conférence qui est acceptée [Collart-Dutilleul et al. 2020].

# Bibliographie


[Adepa 1981] Adepa. Guide d'Etude des Modes de Marches et d'Arrêts (GEMMA).

[Agence ferroviaire européenne] EuRailCheck. [Online]. Available : https ://es.fbk.eu/projects/eurailcheck/index.php

[Agerwala et Flynn] T. Agerwala et M. Flynn. Comments on capabilities, limitations and "correctness" of Petri nets. Proceedings of the 1st annual symposium on Computer architecture, pages 81–86, New York, NY, USA, 1973.

[Amraoui et Mesghouni 2014] A. El Amraoui, et Khaled Mesghouni. Colored Petri Net Model for Discrete System Communication Management on the European Rail Traffic Management System (ERTMS) Level 2. UKSim-AMSS 16th International Conference on Computer Modelling and Simulation, pp. 247-252, 2014.

[Asarin et al. 2000] E. Asarin, O. Bournez, T. Dang, O. Maler, et A. Pnueli. Effective synthesis of switching controllers for linear system. In Proc. IEEE, vol. 88, pp 1011-1025, 2000.

[Ben Ayed et al. 2014] R. Ben Ayed, S. Collart Dutilleul, P. Bon, A. Idani, et Y. Ledru. B Formal Validation of ERTMS/ETCS Railway Operating Rules. 4th International ABZ 2014 Conference, p. 124-129, 2014.

[Beugin et Marais 2012] J. Beugin, et J. Marais. Simulation-based evaluation of dependability and safety properties of satellite technologies for railway localization. Transportation Research Part C : Emerging Technologies, vol. 22, pp. 42–57, 2012.

[Boin et al. 2014] A. Boin, M. Rhinhard, et M. Ekengren. Managing transboundary crises : The emergence of European Union capacity. Journal of Contingencies and Crisis Management 22 (3), pp. 131 - 142, 2014.

[Bon et Collart-Dutilleul 2013] P. Bon, et S. Collart-Dutilleul. From a Solution Model to a B Model for Verification of Safety Properties. Journal of Universal Computer Science, vol. 19, issue 1, pp. 2-24, 2013.

[Boudi et al. 2015] Z. Boudi, E.M. El Koursi, S. Collart-Dutilleul. From Place/Transition Petri nets to B abstract machines for safety critical systems. In Proc. 9th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes SAFEPROCESS 2015, Paris, 2015.

[Boudi et al. 2017] Z. Boudi, R. Ben-Ayed, E.M. El Koursi, S. Collart-Dutilleul, T. Nolasco, et M. Haloua. A CPN/B method transformation framework for railway safety rules formal validation. European Transport Research Review, 13 (9), 15p, 2017.

[Bougacha et al. 2019] R. Bougacha, A. Ait Wakrime, S. Kallel, R. Ben Ayed, et S. Collart-Dutilleul. A Model-based Approach for the Modeling and the Verification of Railway Signaling System. In Proc. the 14th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE 2019), P. 367-376, 2019.

[BS 11200 :2014] British Standards Institution : BSI 11200 :2014. Crisis management. Guidance and good practice. London : BSI Standards Limited, 2014.

[Bundesamt für Verfassungsschutz 2008] Bundesamt für Verfassungsschutz. Baustein ÜA4 Krisenmanagement. The domestic intelligence service of the Federal Republic of Germany, 2008.

[Cassandras et Lafortune 2008] C. G. Cassandras, et S. Lafortune. Introduction to Discrete Event Systems, 2nd ed. New York : Springer, 2008.

[Checkland 1999] P. B. Checkland. Systems Thinking, Systems Practice. Chichester, UK : John Wiley and Sons Ltd, 1999.

[Chutinan et Krogh 2003] A.Chutinan, et B. Krogh. Computational techniques for hybrid system verification. Automatic Control, IEEE Transactions on, vol. 48, no. 1, pp. 64 – 75, 2003.

[Collart-Dutilleul 2013] S. Collart-Dutilleul. Operating rules and interoperability issues in high speed lines. Keynote speech, ICIRT 2013, Beijing, china, Aug.30-Sept.1, 2013.

[Collart-Dutilleul et al. 2020] S. Collart-Dutilleul, H. Kadri, P. Bon, G. Mykoniatis, et S. Ben Ahmed. Security and safety integrated approach for multimodal-hubs crisis management : a railway and airway proposition. Accepté à Transport Research Arena (TRA2020). Helsinki, Finland, 2020.

[CPN Tools] CPNTools Homepage, accessed on September 3, 2019. [Online]. Available : http ://www.cpntools.org/.

[Dhahbi et al. 2012] S. Dhahbi, A. Abbas-Turki, et A. El Moudni. On the ERTMS Level 2 degraded mode : colored Petri net model for discrete point positioning system. IEEE Joint Rail Conference, JRC 2012, Philadelphia, 2012.

[DeLaurentis 2005] D. DeLaurentis. Understanding Transportation as a System-of-Systems Design Problem. 43rd AIAA Aerospace Sciences Meeting, Reno, Nevada, January 10-13, 2005.

[DoD 2008] Ministère de la défense américain (DoD). System of Systems Engineering. Defense Acquisition Guidebook (DAG), Washington, DC : Pentagon, 2008.

[DRMKC] Disaster Risk Management Knowledge Centre (DRMKC). http ://drmkc.jrc.ec.europa.eu/ [Accessed 10.08.2019]

[ECDGET 2006] European Commission and Directorate-General for Energy and Transport (ECDGET). ERTMS–delivering flexible and reliable rail traffic : a major industrial project for Europe. Office for Official Publications of the European Communities, 2006.

[EERC] European Emergency Response Capacity (EERC). https ://ec.europa.eu/echo/what/civil-protection/mechanism_en [Accessed 10.08.2019]

[El Ghadouali et al. 2013] A. El Ghadouali, O. Kamach, et B. Amami. Safe switching of discrete events systems : Application to operating mode management. In Proc. 2013 International Conference on Industrial Engineering and Systems Management, IEEE - IESM 2013, 2013.

[EPISECC] Project Establish a Pan-European Information Space to Enhance seCurity of Citizens (EPISECC). D2.2. Standards, approaches & good practices in cross-border PPDR management. [Project report, Online]. Available : https ://www.episecc.eu/node/78. [Accessed 10.08.2019]

[ERCC] Emergency Response Coordination Centre (ERCC). https ://ec.europa.eu/echo/what/civil-protection/emergency-response-coordination-centre-ercc_en [Accessed 10.08.2019]

[Eusgeld et al. 2011] I. Eusgeld, C. Nan, et S. Dietz. System-of-systems approach for interdependent critical infrastructures. Reliability Engineering & System Safety, vol. 96, no. 6, pp. 679–686, 2011.

[Fabian et Hellgren 1998] M. Fabian, et A. Hellgren. PLC-based implementation of supervisory control for discrete event systems. In IEEE 37th Conference on Decision and Control, vol. 3. Piscataway, NJ : IEEE, pp. 3305–3310, 1998.

[Faraut et al. 2009] G. Faraut, L. Piétrac, et N. Niel. Formal approach to multimodal control design : Application to mode switching. IEEE Transactions on Industrial Informatics 5 (4), pp. 443-453, 2009.

[Faraut et al. 2011] G. Faraut, L. Piétrac, et N. Niel. Equivalence of behaviors between centralized and multi-model approaches. IEEE Conference on Automation Science and Engineering, CASE 2011, Trieste, Italy, pp.32-38, 2011.

[Florin et Natkin 1985] G. Florin et S. Natkin. Les réseaux de Petri stochastiques. Tecniques et Sciences Informatiques, 4(1), pages 143–160, 1985.

[Genrich 1991] H. J. Genrich. Predicate/ Transition nets. High-LevelPetri Nets : Theory and Application, pages 3-43. Springer-Verlag, 1991.

[Guiot et al. 2009] B. Guiot, P. Winter, et International Union of Railways. Compendium on ERTMS : European Rail Traffic Management System. Eurail Press, 2009.

[Hamani et al.2004] N. Hamani, N. Dangoumau, et E. Craye. A formal approach for reactive mode handling. IEEE international conference on Systems, Man and Cybernetics, SMC04, pp. 4306-4311, 2004.

[Hermanns et al. 2005] H. Hermanns, D. N. Jansen, et Y. S. Usenko. From StoCharts to MoDeST : a comparative reliability analysis of train radio communications. In Proc. the 5th international workshop on Software and performance, WOSP '05. New York, USA : ACM Press, pp. 13–23, 2005.

[Herranz et al. 2011] A. Herranz, G. Marpons, C. Benac, et J. Marino. Mechanising the Validation of ERTMS Requirements and New Procedures. In 9th World Congress on Railway Research, Lille, France, p. 33, 2011.

[Huynh et Osmundson 2006] T V. Huynh, et J S. Osmundson. A Systems Engineering Methodology for Analyzing Systems of Systems Using the Systems Modeling Language (SysML). In 2nd System of Systems Engineering Conference, 2006.

[INCOSE 2012] International Council on Systems Engineering (INCOSE). In Systems Engineering Handbook : A Guide for System Life Cycle Processes and Activities. version 3.2.2. San Diego, CA, USA, 2012.

[ISO/IEC. 2002] International Organisation for Standardisation / International Electrotechnical Commissions (ISO/IEC). High-level Petri nets–Concepts, definitions, and graphical notation. Final Draft International Standard 15909, version 4.7.1, 2002.

[ISO/IEC/IEEE 2015] ISO/IEC/IEEE 15288 :2015. Systems and Software Engineering – System Life Cycle Processes. Geneva, Switzerland : International Organisation for Standardisation / International Electrotechnical Commissions / Institute of Electrical and Electronics Engineers, 2015.

[ISO 22316 :2017] International Organization for Standardization : ISO 22316 :2017. Security and resilience. Organizational resilience. Principles and attributes. Geneva : n.a., 2017.

[ISO 22320 :2018] International Organization for Standardization : ISO 22320 :2018. Societal security – Emergency management – Requirements for incident response. Geneva : n.a.

[Jabri et al. 2010] S. Jabri, E. M. El Koursi, et E. Lemaire. European railway traffic management system validation using UML/Petri nets modelling strategy. European Transport Research Review 2(2), pp. 113-128, 2010.

[Jamshidi 2008] M. Jamshidi. Systems of Systems Engineering : Principles and Applications. Taylor & Francis, 2008.

[Jensen 1997] K. Jensen. Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use. vol. 1. Monographs in Theoretical Computer Science, Springer-Verlag, 1997.

[Jensen et al. 2007] K. Jensen, L.M. Kristensen, et L. Wells. Coloured Petri nets and CPN Tools for modelling and validation of concurrent systems. International Journal on Software Tools for Technology Transfer, pp. 213–254, 2007.

[Jensen 2013] K. Jensen. Coloured Petri Nets : Basic Concepts, Analysis Methods and Practical Use. Springer Science & Business Media : Berlin, Germany, 2013.

[Kadri 2010] H. Kadri. Gestion des Modes de Fonctionnement des Systèmes à Evénements Discrets. Mastère de recherche, Institut Supérieur d'Informatique (ISI), Tunis, octobre 2010.

[Kadri et al. 2013] H. Kadri, S. Zairi, et B. Zouari. Global Model For The Management Of Operating Modes In Discrete Event Systems. In Proc. the 6th IFAC Conference on Management and Control of Production and Logistics, MCPL 2013, IFAC Proceedings Volumes ; Volume 46, Issue 24, pp. 420-426, September 2013.

[Kadri et al. 2014] H. Kadri, S. Collart-Dutilleul, et Z. Zouari. Crossing Border in the European Railway System : Operating Modes Management by Colored Petri Nets. In Proc. the 10th Symposium on Formal Methods for Automation and Safety in Railway and Automotive Systems, FORMS/FORMAT 2014, Technische Universitat Braunschweig, pp. 244-252, 2014.

[Kamach 2004] O. Kamach. Approche multi-modèle pour les systèmes à événements discrets : application à la gestion des modes de fonctionnement. Thèse de doctorat, INSA de Lyon, décembre 2004.

[Komenda et al. 2008] J. Komenda, J.H. Van Schuppen, B. Gaudin, et H. Marchand. Supervisory control of modular systems with global specification languages. Automatica, 44(4) :1127–1134, avril 2008.

[Lalouette et al. 2010] J. Lalouette, R. Caron, F. Scherb, N. Brinzei, J. Aubry, et O. Malassé. Evaluation des performances du système de signalisation ferroviaire européen superpose au système français, en présence de défaillances. In Lamda-Mu'2010, vol. 2, La Rochelle, France, pp. 2–9, 2010.

[Leal et al. 2012] A. B. Leal, D. L. L. Cruz, et M. S. Hounsell. PLC-based implementation of local modular supervisory control for manufacturing systems. In Manufacturing System, F. A. Aziz, Ed. Rijeka, Croatia : InTech, pp. 159–182, 2012.

[Lévy-Bencheton et Darra 2015] C. Lévy-Bencheton, et E. Darra. Cyber Security and Resilience of Intelligent Public Transport. Good practices and recommendations. Athens : European Union Agency for Network and Information Security (ENISA), 2015.

[Liu et Darabi 2002] J. Liu, et H. Darabi. Ladder logic implementation of Ramadge-Wonham supervisory controller. In IEEE 6th International Workshop on Discrete Event Systems. Piscataway, NJ : IEEE, pp. 383–389, 2002.

[Mahulkar et al. 2009] V. Mahulkar, S. McKay, D.E. Adams, et A.R. Chaturvedi. System-of-systems modeling and simulation of a ship environment with wireless and intelligent maintenance technologies. IEEE Transactions on Systems, Man, and Cybernetics-Part A : Systems and Humans, 39(6), pp. 1255-1270, 2009.

[Maier 1998] M.W. Maier. Architecting principles for systems-of-systems. Syst.Eng., vol. 1, no. 4, pp. 267–284, 1998.

[Ministère fédéral de l'intérieur de l'Allemagne 2009] Ministère fédéral de l'intérieur de l'Allemagne. National Strategy for Critical Infrastructure Protection. Berlin : n.a., 2009.

[Mortensen 2001] K. H. Mortensen. Efficient Data-Structures and Algorithms for a Coloured Petri Nets Simulator. In Proc. of Third Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools, 2001.

[Nanayakkara et al. 2009] T. Nanayakkara, M. Jamshidi, et F. Sahin. Intelligent Control Systems with an Introduction to System of Systems Engineering. Hoboken, USA : CRC Press, 2009.

[Nourelfath 1997] M. Nourelfath. Extension de la théorie de la supervision à la surveillance et à la commande des systèmes à événements discrets : application à la sécurité opérationnelle des systèmes de prodcution. Thèse de doctorat, INSA de Lyon, 1997.

[Papatheodorou et al. 2014] K. Papatheodorou, N. Klimis, B. Margaris, K. Ntouros, K. Evangelidis, et A. Konstantinidis. An overview of the EU actions towards natural hazard prevention and management : current status and future trends. Journal of Environmental Protection and Ecology 15 (2), pp. 433 - 444, 2014.

[Peter 2011] P. Stanley. ETCS for Engineers. DW Media Group GmbH/Eurailpress, 2011.

[Petri 1962] C. A. Petri. Fundamentals of a Theory of Asynchronous Information Flow. IFIP Congress, pp. 386-390, 1962.

[Pietrac et al. 2002] L. Pietrac, S. Chafik, et L. Regimbal. Application de la théorie de la supervision : un exemple de conception de programmes d'api. Conférence Internationale Francophone d'Automatique (CIFA), Nante, France, 2002.

[Pinheiro et al. 2015] L. P. Pinheiro, Y. K. Lopes, A. B. Leal, et R. S. U. Rosso. Nadzoru : A software tool for supervisory control of discrete event systems. In Proc. of the 5th International Workshop on Dependable Control of Discrete Systems (DCDS), vol. 5, 2015.

[Qiu et al. 2014] S. Qiu, M. Sallak, W. Schön, et Z. Cherfi-Boulanger. Modeling of ERTMS Level 2 as an SoS and Evaluation of its Dependability Parameters Using Statecharts. IEEE SYSTEMS JOURNAL, vol. 8, no. 4, pp. 1169–1181, December 2014.

[Qiu 2014] S. Qiu. Modèles graphiques de l'évaluation de Sûreté de Fonctionnement et l'analyse des risques des Systèmes de Systèmes en présence d'incertitudes. Thèse de doctorat, Université de Technologie de Compiègne, décembre 2014.

[Ramadge et Wonham 1987] P. J. Ramadge, et W. M. Wonham. Supervisory control of a class of discrete event process. SIAM J. Control and Optimization, vol. 25, no. 1, pp. 206–230, 1987.

[Ramadge et Wonham 1989] P. J. Ramadge, et W. M. Wonham. The control of discrete event systems. In Proc. of the IEEE, vol. 77, no. 1, pp. 81–98, 1989.

[Ramdas et al. 2010] V. Ramdas, T. Bradbury, S. Denniss, D. Chapman, R. Bloomfield, et D. Fisher. ERTMS Level 3 Risks and Benefits to UK (TRL PPR PPCA09094). Transport Research Laboratory, Tech. Rep., 2010.

[Rudie et Wonham 1992] K. Rudi, et W.M. Wonham. Think globally, act locally : decentralized supervisory control. IEEE Transactions on automatic control, 37 :1692–1708, 1992.

[Shenhar et Sauser 2009] A.J. Shenhar, et B. Sauser. Systems Engineering Management : The Multidisciplinary Discipline. In Handbook of Systems Engineering and Management, second edition, Edited by : A.P. Sage, and W.B. Rouse, John Wiley & Sons, pp. 117-154, 2009.

[Silva et Queiroz 2010] Y. Silva, et M. Queiroz. Formal synthesis, simulation and automatic code generation of supervisory control for a manufacturing cell. ABCM Symposium Series in Mechatronics - Vol. 4, pp. 418–426, 2010.

[Sun et al. 2014] P. Sun, S. Collart-Dutilleul, et P. Bon. Formal modelling methodology of French Railway Interlocking System via HCPN. Transport Research Arena (TRA 2014), Paris, 2014.

[Sun 2015] P. Sun. Model based system engineering for safety of railway critical systems. Thèse de doctorat, Ecole centrale de Lille, juillet 2015.

[Sun et al. 2015] P. Sun, P. Bon, et S. Collart-Dutilleul. A Joint Development of Coloured Petri Nets and the B Method in Critical Systems. Journal of Universal Computer Science, vol. 21, issue 12, pp. 1654-1683, 2015.

[Surminski et al. 2017] S. Surminski, J. Aerts, D. Alexander, D. Di Bucci, R. Mechler, J. Mysiak, et E. Wilkinson. Prevention and mitigation : avoiding and reducing the new and existing risks. In : K. Poljansek, M. Marin Ferrer, T. De Groeve, I. Clark (Eds.) : Science for disaster risk management 2017 : knowing better and losing less. Luxemburg : Publications Office of the European Union, pp. 449 - 464, 2017.

[Tomlin et al. 2003] C. Tomlin, I. Mitchell, A. Bayen, et M. Oishi. Computational techniques for the verification of hybrid systems. In Proc. the IEEE, vol. 91, no. 7, pp. 986– 1001, 2003.

[UIC-Security Division 2017] UIC-Security Division (eds.) ; Recommendations for Crisis Management. Paris : International Union of Railways (UIC), 2017.

[UN-SPIDER] United Nations Space-based information for Disaster Management and Emergency Response : http ://www.un-spider.org/ [Accessed 10.08.2019]

[UNISIG 2008] UNISIG, ERTMS Users Group. Subset026, System Requirements Specification (SRS). Chapter 4 Modes and Transitions, version 3.0.0, 2008.

[UNISIG 2009] UNISIG. Safety Requirements for the Technical Interoperability of ETCS in Levels 1 and 2. 2009.

[Vademecum – Civil Protection]
http ://ec.europa.eu/echo/files/civil_protection/vademecum/ [Accessed 10.08.2019]

[Vernez et Vuille 2009] D. Vernez, et F. Vuille. Method to assess and optimise dependability of complex macro-systems : Application to a railway signalling system. Safety Science, vol. 47, no. 3, pp. 382–394, 2009.

[Wickramasinghe et al. 2008] N. Wickramasinghe, S. Chalasani, R.V. Boppana, et A. M. Madni. Healthcare System of Systems. System of Systems Engineering Innovations for the 21st Century, (M. Jamshidi, Ed.), John Wiley Series on Systems Engineering, New York, 2008.

[Wonham et Ramadge 1988] W. M. Wonham, et P. J. Ramadge. Modular supervisory control of discrete event system. Mathematics of control, signals and systems, vol. 1, no. 1, pp. 13–30, 1988.

[Wonham 2000] W.M. WONHAM. Supervisory control of discrete-event systems : a tutorial introduction. Tutorial lecture, EAIT, 2000.

[Wonham 2003] W.M WONHAM. Supervisory control theory : models and methods. ATPN – Workshop on Discrete Event Systems Control, 24th International Conference on Application Theory of Petri Nets, ATPN 2003, Eindhover, The Netherland, pp. 1-14, Juin 2003.

[Zimmermann et Hommel 2003] A. Zimmermann, et G. Hommel. A Train Control System Case Study in Model-Based Real Time System Design. In Proc. the International Parallel and Distributed Processing Symposium (IPDPS'03), vol. 00, no. C. Washington, D.C., USA : IEEE Computer Society, 2003.

[Zhong et Wonham 1990] H. Zhong, et W.M. Wonham. On the consistency of hierarchical supervision in discrete-event systems. Automatic Control, IEEE Transactions on, 35(10) :1125–1134, october 1990.

## Gestion des modes de systèmes à événements discrets : application au passage de frontière sous ERTMS

**Résumé :** Les travaux proposés dans ce mémoire présentent une démarche de conception appliquée à la gestion des modes de fonctionnement pour les Systèmes-de-Systèmes (SdSs). Les SdSs sont des grands systèmes dynamiques complexes constitués d'un ensemble de systèmes qui interagissent entre eux en vue de réaliser un objectif commun. La problématique de la conception de ces SdSs porte principalement sur la conception des modes, sur leurs commutations et les relations intersystèmes. Un mode de fonctionnement est un comportement spécifique du système pendant une période de fonctionnement et engageant un ensemble réduit de composants. L'objectif de cette thèse est de proposer une approche de conception sûre des SdSs. Pour réaliser cet objectif, nous utilisons l'approche multi-modèle qui permet de décrire le comportement du système dans un mode donné et la théorie de contrôle par supervision qui permet de concevoir des modèles sûrs par construction. Nous proposons d'abord une démarche de conception des systèmes complexes à plusieurs étapes séparant ainsi les différentes études de conception. Ensuite, nous présentons une généralisation pour couvrir les SdSs. Dans la partie applicative, nous considérons l'ERTMS Niveau 2 comme un SdS et nous résolvons le problème du franchissement de la frontière grâce à la gestion de modes. Enfin, dans une optique de généralisation de notre approche, nous appliquons nos résultats aux systèmes de gestion de crises coopératifs.
**Mots-clefs :** Système-de-systèmes, Théorie de contrôle par supervision, Gestion des modes de fonctionnement, Réseau de Petri haut niveau, ERTMS.

## Mode management in discrete event system : application to border crossing under ERTMS

**Abstract :** The work presented in this thesis concerns a hierarchical design approach applied to the operating modes management of Systems-of-Systems (SoSs). SoSs are large, complex systems whose components are themselves systems which interact with each other to achieve a common goal, and for which the unforeseen failures of a one system can have some serious consequences on the performance of the whole SoS. The design of these SoSs focuses mainly on the operating mode design, on their switching and on inter-system dependencies. An operating mode is a specific behavior of the system during a period of operation and engaging a reduced set of components. The aim of our works is to propose a safe SoS modeling approach. To achieve this goal, we use the Supervisory Control Theory (TCS), which computes safe models in which the requirements are respected, and the multi-model approach that describes the behavior of the system in a given operating mode. The first study proposes a complex systems design approach by construction. The second study presents a generalization of the first one to cover SoSs. In the application section, we consider ERTMS Level 2 as an SoS and we solve the cross-border problem. Finally, in order to generalize our approach, we apply our results to cooperative crisis management systems.
**Keywords :** System-of-systems, Supervisory control theory, Operating modes management, High level Petri net, ERTMS.