Systèmes à événements discrets dans l'algèbre des dioïdes et l'algèbre conventionnelle

Philippe Declerck

UNIVERSITÉ D'ANGERS
ISTIA

Année 2011

Systèmes à événements discrets dans l'algèbre des dioïdes et l'algèbre conventionnelle

Habilitation à Diriger des Recherches

Mémoire scientifique

Présenté le : 2 décembre 2011
à : Angers

par : Philippe Declerck

Maître de Conférences, 61ème section, à l'Université d'Angers

Laboratoire d'Ingénierie des Systèmes Automatisés EA 4094
62, Avenue Notre Dame du Lac 49000 ANGERS

e-mail : philippe.declerck@univ-angers.fr

Table des matières

Chapitre 1

Présentation générale

L'objectif de ce chapitre est de donner une image globale des différents travaux ainsi que le contexte.

1.1 Introduction générale

1.1.1 Contexte

Portant essentiellement sur les systèmes à événements discrets, ce mémoire a été développé au laboratoire Lisa (Laboratoire d'Ingénierie des Systèmes Automatisés - EA4094 -) qui a été créé en 1990 au sein de l'Université d'Angers. A partir de quelques enseignants/chercheurs, la croissance continue du Lisa se traduit actuellement par un effectif de 33 enseignants-chercheurs répartis sur deux axes (Modèles et Systèmes Dynamiques et, Signal-Image) et différents thèmes (http ://www.istia.univ-angers.fr/LISA/). Le contexte définie par cette structure dynamique n'est donc pas celui d'un grand laboratoire disposant de moyens importants (doctorants, personnel technique et administratif,....).

Mon intérêt pour l'étude des systèmes à événements discrets a débuté lors de mes cours de DEA à l'Université de Lille I et a continué aprés ma nomination en tant que Maître de Conférences à l'Université d'Angers en 1993. Résultats d'initiatives individuelles, quelques thèmes semblaient émerger au Lisa mais étaient encore à l'état embryonnaire. Le contexte était celui d'une petite structure de 6/7 enseignants/chercheurs où par exemple, les travaux pratiques d'automatique étaient à monter (une partie du matériel électronique était fait maison et nous devions parfois le réparer car nous n'avions pas de technicien). Sur un point de vue recherche, la seule publication en algèbre (max, +) était une conférence internationale de bonne qualité. La deuxième branche du Lisa portant sur le traitement du signal et l'imagerie médicale n'était pas encore présente. Dans ce cadre en devenir, mon choix a alors été déterminé par la richesse des réseaux de Petri et l'élégance mathématique de l'algèbre des dioïdes comme l'algèbre (max, +). C'était de plus cohérent avec la théorie des graphes que j'avais mis en oeuvre dans ma thèse où je recherchais non le plus grand/petit chemin comme dans l'étoile de Kleene, mais la présence de chemins particuliers.

La thèse portait sur les grands systèmes complexes du type centrale nucléaire/raffinerie dans le cadre d'un contrat avec le centre de recherche de Chatou de l'EDF. Le thème était la détection de défaillances des systèmes non discrets mais continus. Le côté rigoureux de l'algèbre (max, +) où on peut retrouver les notions de vecteurs/valeurs propres et une certaine analogie avec l'automatique classique est indéniablement séduisant même si le niveau d'exigence scientifique en ce domaine est loin d'être négligeable. Certains peuvent même considéré que l'algèbre (max, +) fait partie des mathématiques appliquées car elle a été et est actuellement l'objet d'études de mathématiciens comme M.Gondran et M.Minoux avec leur livre fameux sur la théorie des graphes avec un chapitre sur les dioïdes [12], R.A.Cuninghame-Green et P.Butkovic de l'Université de Birmingham,.... De même, la nécessité d'appliquer des résultats en particulier en commande prédictive, m'a conduit au cours de ces dernières années, à concevoir des algorithmes et m'intéresser à leurs temps de calcul. Développer des compétences en informatique ces dernières années a été un autre enrichissement par rapport à ma formation initiale universitaire en grande partie en électronique et automatique.

Ce mémoire est ainsi le résultat de ces centres d'intérêts mathématiques et informatiques dans le contexte d'une structure donnée, le Lisa. Les chapitres suivants montreront une certaine diversité des thèmes scientifiques que j'ai développés dans différentes algèbres. Ils s'insèrent cependant dans la démarche de l'automaticien (de la modélisation à la commande) avec le souci d'avoir une cohérence mathématique la plus forte possible (dioïdes, réseaux de Petri). Une prise en compte mathématique du problème autant qu'il est possible, facilite la réutilisation future des travaux. Une préocupation de l'automaticien est aussi de développer une théorie sans s'y perdre et de prouver l'applicabilité des démarches proposées en évitant l'écueil du développement exclusif d'applications. Suivant cet esprit, l'objectif est d'établir un corpus théorique cohérent, ancré dans une réalité pratique et réutilisable par un chercheur de la section 61 du CNU (Génie informatique, automatique et traitement du signal).

En recherche, savoir où aller scientifiquement est essentiel car le domaine des sciences est vaste. Les parties suivantes de ce chapitre ainsi que le dernier chapitre de ce mémoire mettrons en valeur les fils conducteurs et les cheminements scientifiques possibles. Parmi ceux-ci, un choix de thèmes a été réalisé afin de concentrer le temps et l'énergie sur des objectifs précis. Bien sûr, d'autres choix dans un secteur scientiquement riche étaient possibles, tous étant aussi passionnants.

Dans les thèmes suivants et pour différentes algèbres, souvent un fil conducteur partira de l'analyse de l'espace des solutions dans différents travaux pour aller dans la détermination d'une solution optimale pour un critère défini.

1.1.2 Démarche générale

Suivant la démarche de l'automaticien, mes travaux de recherche portent sur la modélisation, l'analyse et la commande de systèmes. Ils peuvent être des systèmes de transport, des systèmes agroalimentaires, des systèmes de production d'objets, des systèmes de production énergétique,...

Ils peuvent donc être complexes en raison de leurs tailles, de leur natures plus ou moins cohérentes. Ceci peut induire un système difficilement exploitable, c'est à dire, difficile à observer et à commander, ce qui peut être producteur de retard dans la production ou être crucial pour certains processus du type centrale nucléaire.

La première étape traite de cette complexité au moyen d'une modélisation qui mettra en valeur un aspect essentiel du processus qui pourra être soit la présence d'inter-connections (image structurale d'un grand système complexe comme une centrale nucléaire : voir ma thèse de doctorat effectué à Lille I), l'aspect principalement continu (automatique continu de mes cours), l'aspect événementiel et temporel (systèmes à événements discrets que j'ai traités au Lisa).

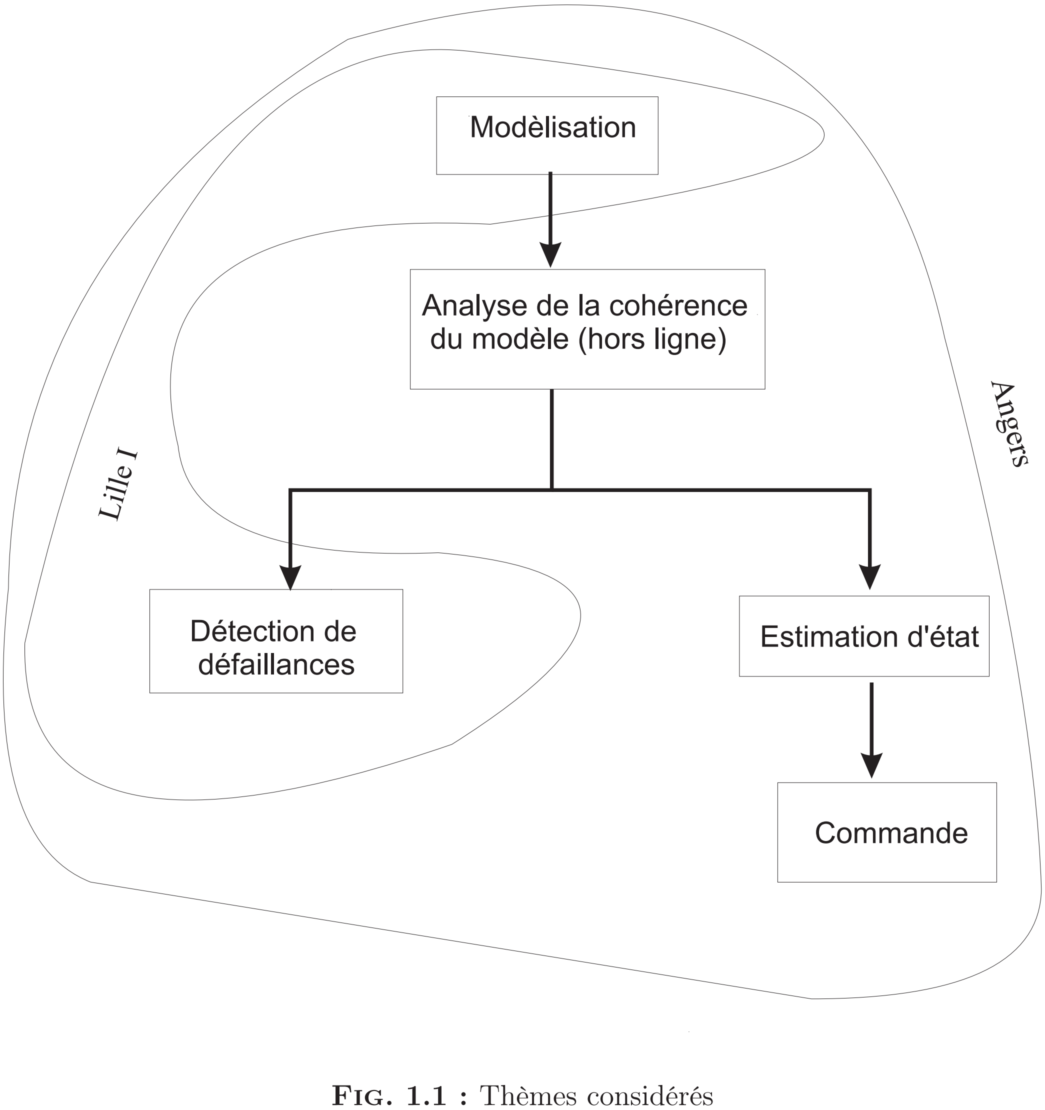

Fig. 1.1 : Thèmes considérés

L'étape suivante analyse le modèle obtenu, celui-ci pouvant présenter des incohérences l'empêchant d'évoluer au cours du temps. On pourra ainsi prévoir qu'au bout d'un certain nombre de répétition de tâches, le système se bloque (et que par exemple, le robot ne soit pas disponible pour retirer à temps un châssis de véhicule baignant dans un bain acide).

Le modèle étant cohérent, le processus peut évoluer vers des comportements non prévus en raison de défaillances et l'incohérence modèle/processus sera mis en évidence par l'analyse des données produites en ligne par le processus.

A partir d'un modèle supposé valide, des problématiques d'analyse, d'estimation d'état et de commande et leurs variantes peuvent ensuite être posées et résolues. Pour ces différents thèmes, la technique dépendra des outils mathématiques qui pourront être la théorie des graphes, l'algèbre des dioïdes ou l'algèbre conventionnelle. En résumé, mes activités scientifiques ont porté sur les deux grandes thématiques de recherche suivantes.

- l'une est sur la détection de défaillance sur de grands systèmes complexes dans le cadre de ma thèse [3] à Lille I et a porté sur les systèmes continus. On détecte les incohérences modèle/processus en utilisant les mesures et les commandes obtenues en ligne et en exploitant la technique de l'espace de parité à travers la décomposition canonique de Dulmage-Mendelsohn.
- l'autre se déroulant actuellement au Lisa à Angers, est sur l'étude des systèmes à événements discrets et a porté sur des thèmes comme l'analyse, l'estimation et la commande. On utilise les treillis, l'algèbre des dioïdes (algèbre (max, +) et ses variantes) mais aussi l'algèbre conventionnelle (programmation linéaire).

Dans ce chapitre, les thèmes seront présentés dans ce chapitre sous une forme synthétique tandis qu'une forme développée sera l'objet des chapitres suivants de ce mémoire. Les différents documents (publications, rapports, mémoires,...) rédigés après l'année 2000 sont accessibles avec le lien *http : //www.istia.univ−angers.fr/˜declerck/* . Les mémoires de DEA/Master recherche et de thèses encadrés sont disponibles sur le site du Lisa *http : //www.istia.univ−angers.fr/LISA/*.

1.2 Travaux à l'Université de Lille I

Développé au sein de l'Université de Lille I, le doctorat [3] portait sur la détection de défaillances qui est basée sur le recoupement d'informations connues venant des capteurs. Le système de surveillance est habituellement composé de trois parties :

- La détection : elle indique la présence ou non d'une défaillance
- L'isolation : Elle donne l'élément défaillant.
- Le diagnostic : Elle fournit le modèle de l'élément défaillant.

Plus précisément, l'objectif du doctorat fixé par Marcel Staroswiecki, était la détection de défaillances de grands systèmes complexes du type centrale nucléaire. La grande difficulté de ce problème, pour le moins ambitieux, était la complexité du processus en raison de sa taille et de sa nature multiple (bi-linéaire, non-linéaire, qualitatif/quantitatif,...).

Face à ce problème, j'ai développé une démarche d'analyse structurale qui a été le point de départ d'une nouvelle thématique dans la détection de défaillances. Cette thématique nouvelle à l'époque a largement survécu à mon changement de région et de thème. Elle est par exemple

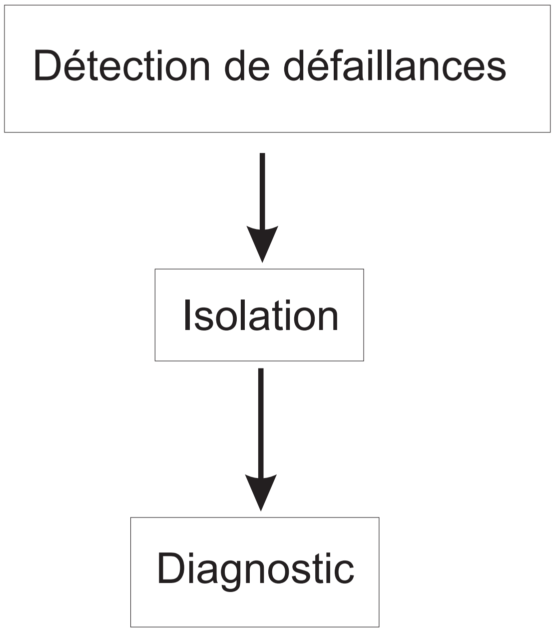

Fig. 1.2 : Détection de défaillances

illustrée par l'habilitation de Vincent Cocquempot en 2004 dont le chapitre 7 du mémoire [2] s'appuie sur mes résultats de thèse et par le thème d'une session invitée lors de la conférence internationale en 2005, 17th IMACS World Congress Scientific Computation, Applied Mathematics and Simulation, qui reprend complètement ce thème.

La technique a été de mettre en valeur la présence d'inter-connections du processus. On remplace ainsi le modèle algèbrique difficilement exploitable par une image structurale qui est un graphe bi-parti fonctions/variables où on considère que les variables inconnues. On effectue un couplage maximal fonctions/variables et on applique ensuite la décomposition canonique de Dulmage-Mendelsohn. L'analyse de la structure mathématique d'une centrale nucléaire montre sous certaines hypothèses, que, pour un couplage *maximal*, certaines parties peuvent être surveillées (plus de fonctions que de variables ; partie SUR dans le dessin), d'autres ne permettent qu'une estimation de l'état (autant de fonctions que de variables pour un *couplage complet* ; partie JUSTE) et enfin que d'autres encore ne permettent aucun traitement (plus de variables que de fonctions ; partie SOUS).

Dans la partie SUR, l'élimination des variables inconnues produit des équations ne comportant que des variables connues, dont l'égalité est testée en ligne, en parallèle avec le processus en fonctionnement. Lorsque l'égalité n'est plus vraie, une défaillance est détectée.

Ce doctorat avait comme support financier un contrat avec la Direction des Etudes et Recherche de Chatou de l'EDF. Ceci a abouti à la rédaction de plusieurs rapports industriels et un logiciel de simulation d'un modèle simplifié d'une centrale PWR 900MW.

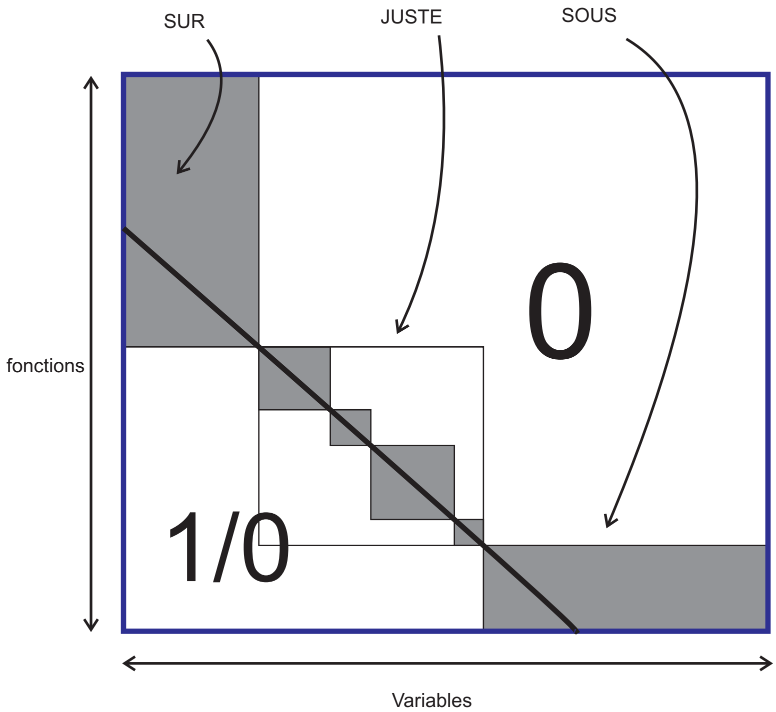

FIG. 1.3 : Décomposition canonique de Dulmage-Mendelsohn : 0 pour pas de lien et 1/0 lien possible ; le couplage maximal est symbolisé par la diagonale ; les carrés au centre sont des structures irréductibles.

1.3 Travaux à l'Université d'Angers

Avant de décrire les travaux effectués à l'Université d'Angers à partir de 93, nous présentons dans cette partie le positionnement mathématique (dioïdes), un positionnement à l'intérieur des réseaux de Petri et les modèles traités.

1.3.1 Positionnement

1.3.1.1 Dioïdes

Il existe différentes structures d'ensembles munis de une ou deux opérations [12]. Nous considérons dans ce mémoire principalement la structure de dioïde qui est un semi-anneau idempotent et l'algèbre (max, +) $(\mathbb{R} \cup \{-\infty\}, \oplus, \otimes)$. C'est un outil permettant de faire de la théorie des graphes, de gérer un système à événements discrets,... Plus connue, nous avons la structure d'anneau avec pour exemple l'algèbre conventionnelle ($\mathbb{R}$,+,x) qui est utilisé dans de nombreux domaines comme l'automatique. Le point commun est donc la structure de semi-anneau qui constitue le socle aux structures de dioïdes et d'anneaux comme le montre la figure 1.4. La différence essentielle est la présence de la symétrie pour la première opération (addition) des anneaux qui est remplacée par l'idempotence pour la première opération (maximum pour l'algèbre (max, +)) des dioïdes. Cette différence est essentielle car il est prouvé que l'ensemble se réduit à l'élément neutre $\{-\infty\}$ de la première opération $\oplus$ si on suppose une structure présentant les deux propriétés. Nous sommes donc à la croisée de deux chemins allant, l'un vers l'automatique des systèmes continus, l'autre allant vers les systèmes à événements discrets. L'analogie "automatique classique/systèmes à événements discrets" est limitée par le tronc commun qui est la structure de semi-anneau, même si les systèmes à événements discrets peuvent s'inspirer avec profit des principes de l'automatique. Sur un point de vue mathématique, cette différence explique pourquoi il n'est pas possible de construire un nouvel ensemble à partir de $\mathbb{R} \cup \{-\infty\}$ comme dans l'algèbre conventionnelle où la technique de symétrisation construit $\mathbb{Z}$ à partir de $\mathbb{N}$. Le lecteur intéressé par ce point important pourra consulter le théorème 3.62 page 129 dans le chapitre "Symmetrization of the Max-Plus Algebra" de [1] et la figure 2 page 34 dans le chapitre "Pré-semi-anneaux, semi-anneaux et dioïdes" de [13]

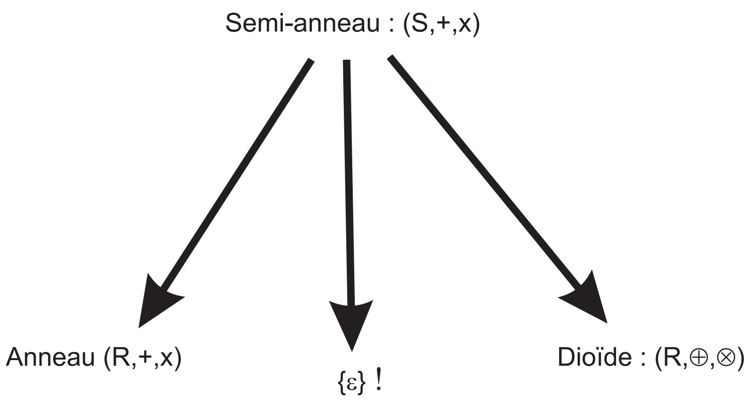

Fig. 1.4 : Deux algèbres : point commun et opposition

Cette vision dichotomique sera rediscutée dans le dernier chapitre portant sur les perspectives.

1.3.1.2 Réseaux de Petri et l'extension du temps

Dans le domaine des systèmes à événements discrets, la classe des réseaux de Petri constitue un support graphique intéressant permettant de modéliser assez facilement de nombreux types de systèmes comme les processus industriels, les systèmes de transport, agroalimentaires,... Nous avons ainsi développé le modèle d'un processus de panification semi-industriel lors d'une collaboration avec Cécile Grémy-Gros du laboratoire Lasquo.

D'autre part, tout processus se déroule dans le temps et présente une évolution avec des événements si nous considérons un systèmes à événements discrets. L'extension du temps aux réseaux de Petri constitue donc également une motivation intéressante. Il existe aussi des exemples qui présentent un caractère concrêt, comme l'arrivée à l'heure en classe des étudiants et de l'enseignant (voir [8]) .

Les réseaux de Petri peuvent être classés par rapport à deux types fondamentaux (voir l'article de T. Murata de 1989 [14]) qui déterminent également deux thèmes de recherches qui regroupent des chercheurs en général différents.

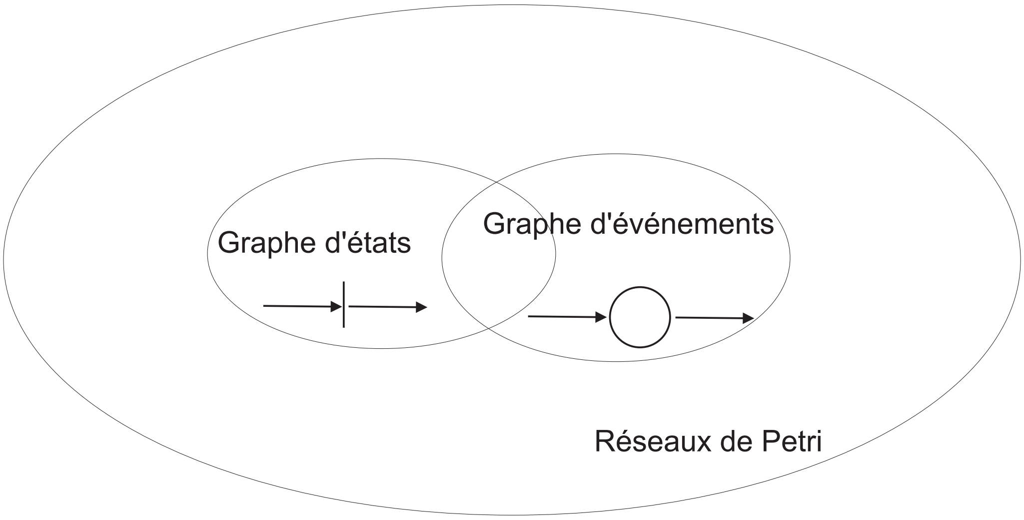

FIG. 1.5 : Réseaux de Petri : les structures clés d'après la figure 25 de T.Murata 1989

- Les graphes d'événements (chaque place a exactement, une transition d'entrée et une transition de sortie) mettent en valeur la synchronisation mais ne présentent pas de choix. Dans ce type de modèle, l'introduction du temps est aisée et il est possible de définir des synchronisations complexes. Nous choisirons ce modèle ou une extension de celui-ci, par la suite.

- Les graphes d'états (chaque transition a exactement, une place d'entrée et une place de sortie) se focalisent sur les choix mais ne contiennent pas de synchronisation. La théorie du langage est bien adaptée. Introduire le temps est souvent peu commode.

- L'intersection des deux ensembles précédents est non vide mais représente des réseaux de Petri ne pouvant modéliser ni la synchronisation, ni le choix.

- L'union des deux ensembles précédents, c'est à dire la considération de modèles prenant des éléments aux deux modèles fondamentaux, est à l'origine de travaux croisés. Cependant, la considération d'une combinaison directe de ces modèles en tenant compte du temps, aboutit rapidement à une explosion combinatoire limitant le domaine d'applications comme dans le Model Checking qui génére une origine de temps à chaque événement (tir de transition). Contrairement à ces études spécialisées sur ce cas complexe, nous considérons dans nos travaux une origine des temps arbitraire mais unique avec une possibilité pour le temps d'être négatif.

1.3.1.3 Extension des modèles et dateurs

Mes travaux se placent en grande partie dans l'action 3 : Extension de la classe des systèmes (max, +) de l'équipe Modèles et Systèmes Dynamiques du laboratoire Lisa (cette action sera moins visible dans le nouveau organigramme du Lisa en 2011).

Dans le domaine des Réseaux de Petri, les graphes d'événements du type P-temporel, T-temporel ou arc-temporel (graphes d'événements à flux temporels ou time stream event graphs) permettent de décrire des phénomènes complexes de synchronisation et de modéliser des systèmes de production, de transport, agroalimentaires ou multimédia sous le point de vue du temps. La trajectoire de l'état n'est pas déterminée par l'itération d'une équation d'état dans l'algèbre (max,+) mais par une itération d'un intervalle qui définit un espace d'évolution. Les bornes sont définies par des fonctions utilisant les opérations minimum, maximum et addition. Ces modèles sont dits du type intervalle. On obtient ainsi des modèles (max,+) mais aussi des modèles (min, max, +).

Notons que les modèles considérés sont le résultat d'un parti pris qui est celui d'utiliser les dateurs, c'est à dire, de repérer chaque événement par une date numérotée et non un compte daté appelé aussi compteur. Ceci a une influence non négligeable sur la modélisation initiale qui est une étape délicate car celle-ci peut aboutir à un modèle peu exploitable. Bien sûr, il est toujours possible de convertir numériquement une trajectoire exprimée dans une forme, dans l'autre. Une conséquence est que les modèles présents dans ce mémoire, décriront des *phénomènes complexes de synchronisation* ceci pour des *structures simples* de réseaux de Petri. On se placera dans l'espace des réels et on utilisera l'algèbre ($\mathbb{R} \cup \{-\infty\}$, max, $+$) et ($\mathbb{R} \cup \{-\infty\}$, min, max, $+$) complétée éventuellement avec $+\infty$. A l'inverse, le compteur permettra de considérer des *structures complexes* et des valuations non unitaires pour des *synchronisations simples*. On sera obligé de définir les compteurs dans les entiers naturels ou relatifs. On considère l'algèbre ($\mathbb{Z} \cup \{+\infty\}, min, +$) ou ($\mathbb{N} \cup \{+\infty\}, min, +$) complétée éventuellement avec $-\infty$.

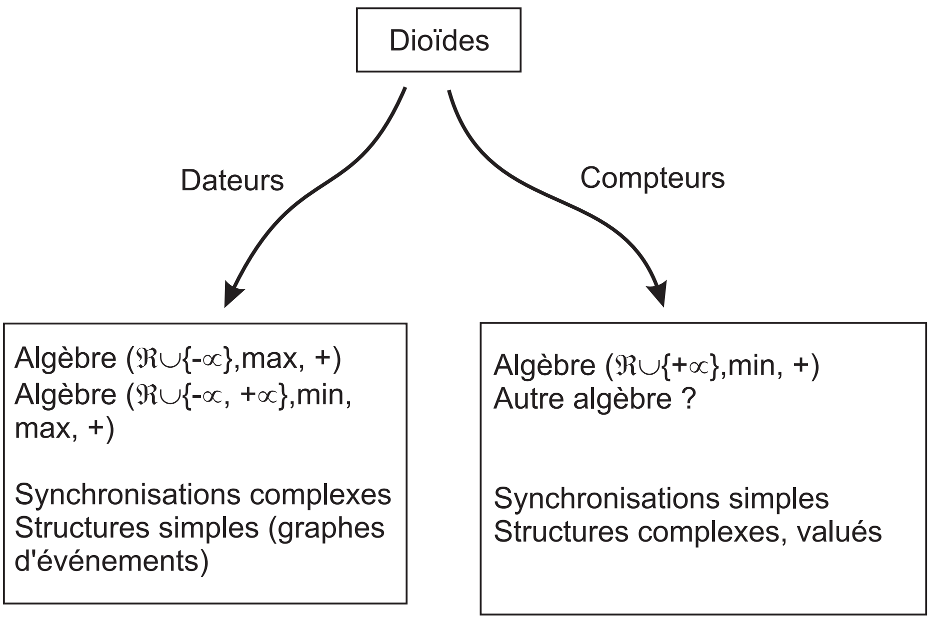

Fig. 1.6 : Dateurs et compteurs

Une comparaison des différents types de graphes d'événements temporels et de modèles du type intervalle seront présentée dans le début des chapitres sur la vivacité et la commande prédictive, respectivement. Une description détaillée des modèles pourra être trouvée dans les mémoires de thèse de Khalid Didi Alaoui [21] et Abdelhak Guezzi [23].

1.3.2 Thèmes et principaux outils utilisés

Les thèmes portant sur ces modèles basés sur les dateurs seront les suivants. L'espace d'évolution peut être vide ce qui soulève le problème de la vivacité temporelle et de la perte de ressources. Un système vivant peut être plus ou moins productif et il est intéressant de connaître son taux de production ou son temps de cycle. L'objectif est également de développer la commande et l'estimation d'état pour ces nouveaux modèles.

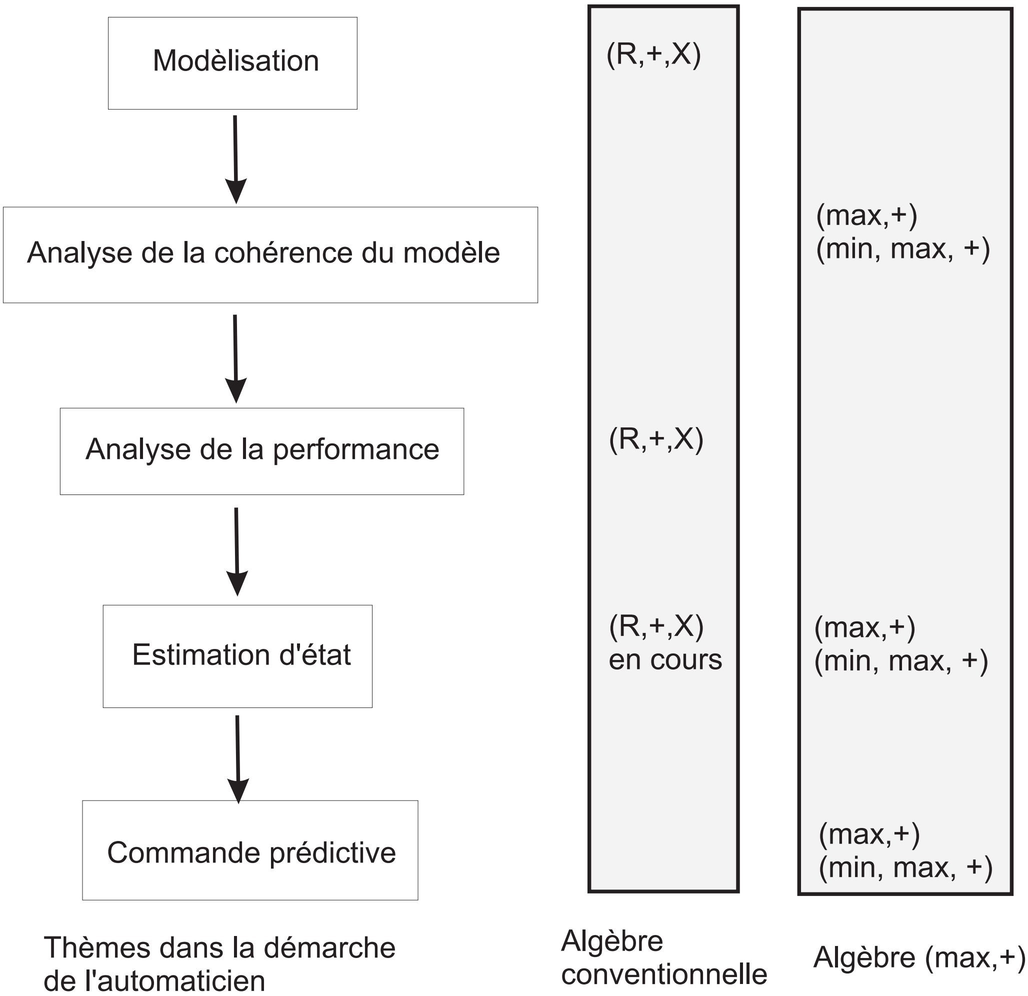

FIG. 1.7 : Dans la démarche de l'automaticien, thèmes traités par rapport aux algèbres.

Ces différents thèmes peuvent être souvent vus comme une résolution du type point-fixe, c'est à dire une résolution itérative des formes $x \leq f(x)$, $x = f(x)$ et $x \leq f(x)$ jusqu'à stabilisation, la fonction f étant monotone. Déjà une définition de f plus précise suggère des problèmes théoriques car l'ajout des propriétés d'homogénéite ($f(x+h) = f(x)+h$ avec $h \in \mathbb{N}$) et de non-expansivité définisse les fonctions topicales dont les fonctions (min, max, $+$) sont une sous-classe.

On utilisera également la programmation linéaire par la suite.

1.3.2.1 Vivacité temporelle des réseaux de Petri

La modélisation peut aboutir à un modèle ne pouvant pas définir de trajectoire sur un horizon infini. L'origine de cette difficulté est un blocage des tirs des transitions pouvant provoquer la mort des jetons si on considère un réseau de Petri P-temporel. Considérant la plus grande trajectoire possible par rapport au nombre d'événements, nous avons développé une technique déterminant cette non-synchronisation des transitions, c'est à dire, le plus grand numéro de l'événement correspondant (article IEEE-TAC en 2011 [10]) ainsi que la date des premières morts de jeton. Nous déterminons aussi le modèle dégradé obtenu après cette perte de jeton exprimant par exemple une perte de ressource.

On s'est également appuyé sur la théorie spectrale développée par J. Cochet-Terrasson, S. Gaubert et J. Gunawardena. La transposition de cette théorie dans le domaine des réseaux de Petri donne une approche pour tester la vivacité temporelle des graphes d'événements plus généraux, du type intervalle (article IEEE-TAC en 2010 [8]) sur un horizon donné.

1.3.2.2 Temps de cycle/taux de production

Choisissant l'algèbre conventionnelle, nous utilisons un modèle de la forme polyédrale $A.x \leq b$ pour évaluer et analyser les performances des graphes d'événements. Nous considérons ici une généralisation du graphe d'événements P-temporel auquel est associé des inter-dépendances entre temps de séjour. Considérant un comportement 1-périodique, l'application d'un Lemme de Farkas permet de tester l'existence du temps de cycle ainsi que la détermination des bornes l'encadrant. Comme cette technique revient à énumérer tous les vecteurs orthogonaux positifs, deux techniques calculant le temps de cycle minimal et maximal, et en étant numériquement plus efficaces, sont développées : la première approche qui sera nommée primale, puisqu'elle se base sur l'écriture primale de la programmation linéaire, permet de trouver un vecteur de dates qui peut être appliqué au système. La deuxième approche duale donne les sous-structures et ses temporisations qui limitent la vitesse de production par des bornes. On utilise alors le théorème de la dualité, bien connue en programmation linéaire, ainsi que le théorème de Stiemke qui permet une simplification de forme.

Différentes parties de cette étude mais limitées aux graphes d'événements P-temporels ont été publiées dans les conférences internationales ICINCO 07 et CIFA 08. En dehors de l'utilisation de théorèmes de la programmation linéaire (classiques mais peu exploités en général dans la communauté réseau de Petri), le point important de cette partie est que le concept de temps de cycle existe encore pour le modèle généralisé alors que sa détermination ne peut plus se réduire à l'analyse classique des circuits dans le réseau de Petri ou dans un graphe associé arcs-sommets qui ne peut plus être défini. Les résultats classiques de l'algèbre (max, +) avec le théorème de Karp qui permet une résolution polynômiale au sens fort, ne peuvent plus s'appliquer.

1.3.2.3 Observateur

L'approche que j'ai développée est une transposition des observateurs classiques de l'automatique classique dans l'algèbre (min, max, +) : on peut donc maintenant considérer des systèmes à événements discrets complexes du type intervalle. Utilisant un horizon glissant, la technique développée opère une estimation du vecteur d'état. La redondance des informations fait que de plus, celles-ci peuvent être recoupées ce qui permet de faire de la détection de défaillances : on retrouve la démarche de l'Espace de Parité que j'ai analysée et développée durant ma thèse de doctorat mais pour des systèmes continus et une algèbre différente. Il a donc fallu changer d'algèbre, développer une technique du type point fixe adaptée et tenir compte de la nature différente du système. Mathématiquement, le problème est celui de la détection d'un espace vide de solution pour un problème posé sous la forme d'un point fixe du type (min, max, +).

Au final, l'approche permet d'estimer le plus grand état possible avec les informations disponibles et de détecter des défaillances ou des changements de modèles imprévus. La localisation (l'isolation) des sous-structures défaillantes est possible. Nous pouvons aussi détecter des variations même petites de temporisations qui peuvent être des réductions ou des augmentations. Ces informations montrant une dégradation du processus permettent de déclencher des opérations de maintenance préventive.

Cela se traduit par un article dans la conférence internationale DCDS'07 [6] ainsi qu'une soumission en cours dans une revue de premier plan.

1.3.2.4 Commande

L'objectif est de synthétiser la commande d'un graphe d'événements temporisé sous le critère classique du juste-à-temps : la commande doit retarder, autant que possible le système, de sorte que la sortie se produise avant une sortie désirée définie sur un horizon fini donné.

L'étude présentée dans la publication Kybernetika 1999 [4] se place dans la situation où le vecteur d'état est inconnu. L'approche choisie n'est pas alors d'estimer le vecteur d'état avec un observateur mais de l'éliminer en générant les équations appelées ARMA (AutoRegressive and Moving Average) par analogie avec l'algèbre standard : on retrouve alors une démarche analogue à celle de l'Espace de Parité en détection de défaillances qui produit des relations analytiques de redondance. Considérant des matrices d'état irréductibles, on pourra calculer la commande moyennant une hypothèse de périodicité de trajectoire sans transitoire. L'utilisation des équations ARMA permet d'opérer sur un horizon glissant fini et ainsi de modifier facilement les paramètres du problème comme la trajectoire désirée.

Une autre étude ajoute des contraintes additionnelles qui correspondent, non à un seul modèle mais à un ensemble de graphes d'événements du type intervalle. On doit alors sortir de l'algèbre (max, +) : le point fixe obtenu a une forme contenant des maximisations, minimisations et additions. On peut utiliser la théorie spectrale pour tester la cohérence de l'ensemble des modèles. Nous avons donné un algorithme qui calcule le plus grand vecteur de commande ainsi que la plus grande trajectoire exactement suivie par le système. L'approche est décrite dans un article IEEE-TAC en 2010 [8].

En se limitant à des contraintes additionnelles du type graphes d'événements P-temporel, nous avons aussi traité le problème de la poursuite de trajectoire sur un horizon glissant. En dehors de la résolution propre de la commande, un problème essentiel est de pouvoir calculer à temps les résultats afin de pouvoir utiliser l'état estimé dans un problème d'estimation ou de pouvoir appliquer la commande dans un problème de commande. Un critère à considérer est la complexité des algorithmes afin de réduire le temps d'exécution des algorithmes. La technique adoptée est de se ramener à une structure de treillis même si le système initial n'a pas cette structure.

Nos simulations montrent que notre technique peut traiter de grands systèmes (90 transitions) sur des horizons importants (50 pas) qui sont glissants à l'infini. Une description de la technique peut être trouvée dans un article d'un livre " Lecture Notes in Control and Information Sciences" en 2009 [7] ainsi que dans un article de la conférence internationale Wodes 2010 [9].

Enfin, nous avons développé ces deux dernières années une technique formelle de commande prédictive dans le sens qu'elle s'appuie sur un calcul en ligne numériquement réduit, contrairement à de nombreux travaux du domaine (méthode Model Predictive Control de Bart de Schutter et Ton van den Boom) : c'est un correcteur de l'espace d'état dans le sens que la commande calculée imposera que le système se place dans un sous-espace tel que la trajectoire générée suive le modèle du processus et les spécifications. Cette étude est l'objet d'une soumission en cours dans une revue de premier plan.

1.4 Présentation des chapitres suivants

Après cette présentation synthétique des thèmes, une description plus développée est l'objet des chapitres suivants. Afin d'obtenir un document de taille raisonnable, nous nous limiterons à l'analyse de la vivacité, l'analyse du temps de cycle et la commande prédictive, la majeure partie des travaux étant accessibles sur le site du Lisa ($http : //www.istia.univ-angers.fr/\tilde{}declerck/$ et $http : //www.istia.univ - angers.fr/LISA/$). Deux algèbres clairement distinctes selon un point de vue mathématique, seront considérées :

- l'algèbre conventionnelle pour le chapitre analyse du temps de cycle

- l'algèbre (max, +) et l'extension l'algèbre (min, max, +) pour les chapitres sur l'analyse de la vivacité et la commande prédictive.

En dehors du fait que les chapitres suivants permettent de diffuser des résultats dans des thèmes centraux à la section 61 et qu'ils s'insèrent dans la démarche de l'automaticien, ils suivent aussi la logique suivante. Les chapitres principalement sur le taux et la commande prédictive participent à une réflexion globale sur la justification de l'algèbre (max, +) et son extension dans l'algèbre (min, max, +) et la programmation linéaire. De même, la nécessité d'avoir des

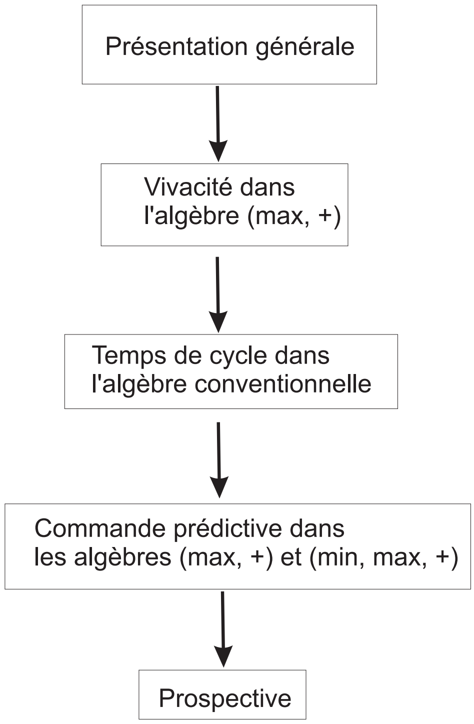

FIG. 1.8 : Chapitres du mémoire dans les différentes algèbres

temps de calcul efficaces pour la plus grande classe de problèmes possibles dans les chapitres vivacité et commande prédictive sur un horizon glissant, motive cette réflexion tout en poussant à s'intéresser au domaine informatique. Le dernier chapitre Prospective donnera une synthèse basée sur ces différents chapitres qui permettra de mieux formaliser cette réflexion et de proposer un projet scientifique.

Disponibles que dans ce mémoire, ces chapitres à l'exception du dernier, suivent en partie des documents déjà rédigés et parfois non encore publiés dans une conférence ou une revue. Le chapitre vivacité est construit à partir d'un rapport technique accessible sur ma page personnelle [16] et d'un article IEEE-TAC en 2011 [10]. Une partie du chapitre sur la commande prédictive avec horizon glissant a été établi à partir des articles présentés à Posta 2009 [7] et Wodes 2010 [9]. Le chapitre Commande prédictive est une version étendue de l'article IEEE-TAC en 2010 [8] comportant de nouveaux exemples ainsi que la considération des conditions initiales. Le chapitre sur le temps de cycle généralise des articles de conférences internationales assez récents en considérant un nouveau modèle étendu et une nouvelle interprétation graphique. Enfin, la technique de calcul de l'étoile de Kleene de la matrice tri-diagonal est non-publié. Dans le dernier chapitre aussi non-publié, une prospective basée sur des travaux actuels et passés sera proposée : en effet, les termes de *directeur de recherche* suppose par définition une vision scientifique d'une recherche future avec des pistes fécondes dans lesquelles son équipe pourra s'engager.

Les chapitres suivants sauf le dernier seront également rédigés en anglais qui est la langue scientifique internationale du moment. Nous donnerons quelques exemples intuitifs qui, tout en facilitant l'accès aux concepts, permettrons de suggérer des résultats plus généraux. Chaque chapitre est quasi-autonome et peut être lu indépendamment des autres.

Bibliographie

[1] F. Baccelli, G. Cohen, G.J. Olsder et J.P. Quadrat, Synchronization and Linearity. An Algebra for Discrete Event Systems, disponible sur http ://maxplus.org, New York, Wiley, 1992.

[2] Vincent Cocquempot, Contribution à la surveillance des systèmes industriels complexes, Habilitation à Diriger des Recherches, Université de Lille I, 2004.

[3] Ph. Declerck, Analyse structurale et fonctionnelle des grands systèmes complexes. Application à une centrale PWR 900 MW, Thèse de doctorat, Université de Lille I, 1991.

[4] Ph. Declerck, Predictability and control synthesis in Timed Event Graphs. Kybernetika, Académie des Sciences de la République Tchèque, vol. 35, n°1, p. 25-38, 1999.

[5] Ph. Declerck, From extremal trajectories to consistency in P-time Event Graphs, disponible sur http ://www.istia.univ-angers.fr/∼declerck/, rapport technique, Lisa, Université d'Angers, 26 septembre, 2007.

[6] Ph. Declerck, Detection of changes by Observer in Timed Event Graphs and Time Stream Event Graphs, disponible sur http ://www.istia.univ-angers.fr/∼declerck, IFAC DCDS'07, Cachan, France, 2007.

[7] Ph. Declerck, A. Guezzi, Trajectory Tracking Control of a Timed Event Graph with Specifications Defined by a P-time Event Graph, POSTA 09 Positive Systems : Theory and Applications, Valence, Espagne, 2-4 septembre, 2009.

[8] Ph. Declerck et M.K. Didi Alaoui, Optimal control synthesis of timed event graphs with interval model specifications, IEEE Transactions on Automatic Control, février, 2010.

[9] Ph. Declerck, Trajectory Tracking Control of a Timed Event Graph with Specifications Defined by a P-time Event Graph : On-line control and Off-line preparation, Wodes 2010, Berlin, Allemagne, 30 Août au 1 septembre, 2010.

[10] Ph. Declerck, From extremal trajectories to consistency in P-time Event Graphs, IEEE Transactions on Automatic Control, 2011.

[11] M.K. Didi Alaoui, Etude et supervision des graphes d'événements temporisés et temporels : vivacité, estimation et commande, Thèse de doctorat, Université d'Angers, disponible sur http ://www.istia.univ-angers.fr/, 2005.

[12] M. Gondran et M. Minoux, Graphes et Algorithmes, John Wiley and Sons, mars 1984.

[13] Graphes, dioïdes et semi-anneaux, Editeur : Tec et Doc, Eyrolles, 2002.

[14] A. Guezzi, Ph. Declerck et J.-L. Boimond, Commande de graphes d'événements temporisés sur un horizon glissant, Journal Européen des Systèmes Automatisés (JESA), vol. 43/7-9, p. 1097-1111, 2009.

[15] A. Guezzi, Modélisation, analyse de performance et commande des systèmes à événements discrets, Thèse de doctorat, Université d'Angers, disponible sur http ://www.istia.univ-angers.fr/, 2010.

[16] T. Murata, Petri Nets : Properties, Analysis and Applications, Proceedings of the IEEE, Vol. 77, No. 4, 1989.

Chapitre 2

Vivacité

Dans cette partie, nous analysons la classe des Graphes d'Evénements P-temporels à travers leurs caractères originaux comme la possibilité de mort de jetons et l'existence de contraintes rétroactives : le présent influence naturellement le futur mais il est aussi nécessaire dans ce modèle, de considérer la présence de contraintes du futur sur le présent. Cette analyse est réalisée au moyen d'une série particulière de matrices au sein de l'algèbre (max, +).

2.1 Introduction

2.1.1 Models

Petri Nets (PNs) with time can express the time behavior of Discrete Event Systems with their specifications. Two main behaviors of the transitions can be distinguished : firing as soon as possible in Timed PNs and firing in given time intervals for Time PNs. Time can be associated with places, transitions and arcs of the PNs. In Time Stream PNs, temporal intervals are associated with arcs outgoing from places and the firing interval of transitions is defined by different semantics ([8] [18]). For Timed PNs, durations can be associated with places (P-Timed PNs) or transitions (T-Timed PNs) and the relevant subclasses are equivalent. For Time PNs, temporal intervals can similarly be associated with places or transitions but the corresponding subclasses (P-Time PNs and T-Time PNs) are fundamentally different. In Time PNs, a temporal interval of firing is associated with each transition enabled by the marking while a temporal interval of availability is associated with each token which enters a place in P-Time PNs. (Chapter 6 in [18])

Moreover, T-Time PNs and P-Time PNs correspond to two different semantics of Time Stream PNs : a T-Time PN is similar to a Time Stream PN for specific intervals, where the semantic of the transitions is "Weak-And" ; The algebraic model of a P-Time Event Graph is identical to a Time Stream Event Graph where the semantic of the transitions is not "Weak-And" but "And" [19].

An algebraic synthesis of the different classes of Time Event Graphs is as follows. Time Event Graphs can be described by a new class of systems called interval systems [9] [13] for which the

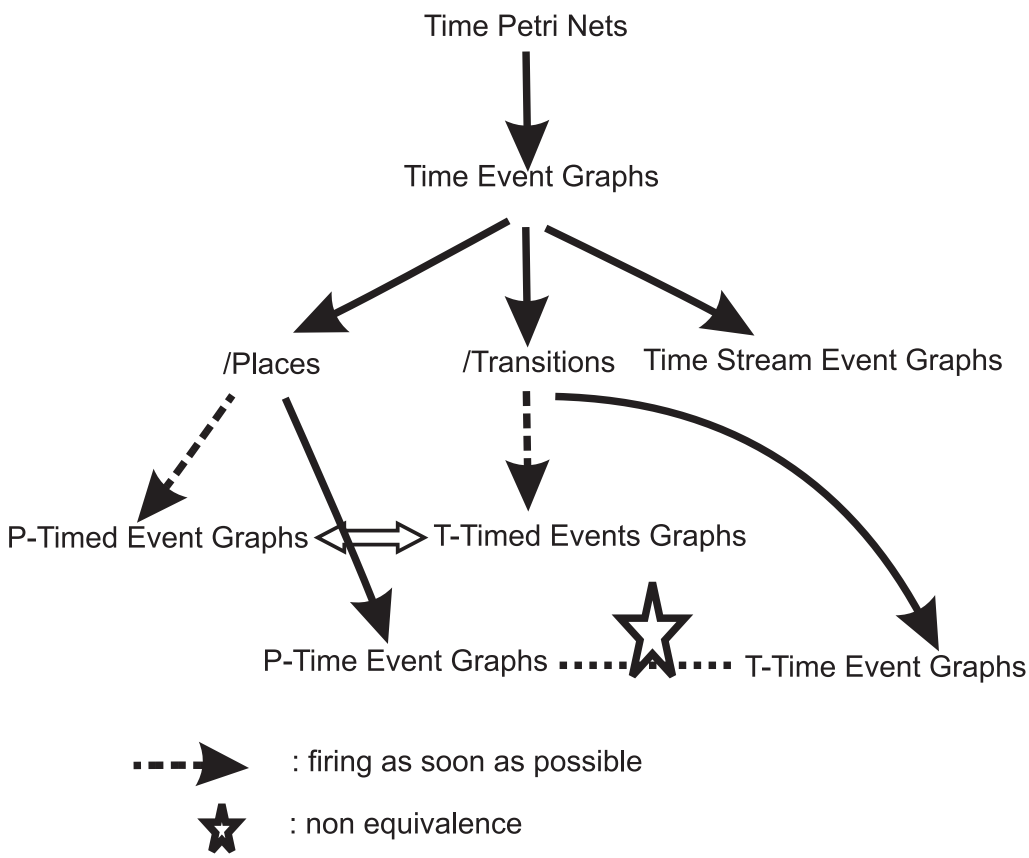

FIG. 2.1 : Time Petri Nets

time evolution is not strictly fixed but belongs to intervals. For interval systems, lower and upper interval bounds depend on the operations of maximization, minimization and addition, in general case. The lower and upper bound constraints of T-time Event Graphs are (max, +) functions while the lower and upper bound constraints of P-time Event Graphs are (max, +) and (min, +) functions, respectively. More details on interval systems can be found in the chapter on Predictive Control.

2.1.2 Physical point of view

P-time Event Graphs define a set of trajectories which follows the specifications given by the model for a nominal behavior. Firing dates of the transitions belong to the relevant time windows. Indeed, P-time Petri nets are convenient tools to model manufacturing systems whose operation times must be included between minimum and maximum values. Applications of P-time Event Graphs can be found in production systems [17], food industry [6] and transportation systems [7]. A practical example is the electroplating lines where the minimum and maximum immersion times guarantee the quality of the chemical treatment [7] : Each chemical treatment must be sufficient otherwise the product will not be ready for the next task or treatment ; On the other hand, each chemical treatment must not be too excessive, otherwise, the product would be damaged. Other practical examples can be given in food industry. In good bakery practice, the dough stays in the fermentation room from three to five hours, the time depending on room temperature and flour or gluten quality. The loaves need to be baked between a minimal and maximal time. If these times are too short or too long (e.g. a synchronization with another operation is not fulfilled), the product will be damaged (bad inner structure and grain in the finished loaf, insufficient or excessive baking). In transportation systems (bus, subway,..) we can consider additional constraints as expected minimum departure frequency which defines maximum time separation between two vehicle departures at a stop. Moreover, in air traffic, a vehicle must respect security distance with the vehicle ahead, which can be expressed by a temporization. In this part, we focus on P-time Event Graphs [21] whose evolution can undergo token deaths which express the loss of resources or parts and failures to meet time specifications.

2.1.3 Objectives

However, even if the underlying graph of the Event Graph is live, the specifications can be too restrictive and some synchronizations cannot follow the desired model, producing the losses of resources. A process is composed of machines, resources, etc. and the issue is to know if they can work together following a schedule for a specified period. More particularly, a question is to know if the different tasks can be sufficiently repeated during a period such that a normal production can be performed without losses of parts. A practical problem can appear when a machine is changed : using a slower/quicker machine can affect the nominal behavior of the complete production line defined by a previous schedule. In this paper, we are interested in avoiding this situation. The relevant notion is the consistency, which can be defined by the existence of a time trajectory following the model. Many approaches as control, simulation, optimization, etc. are based on a model and they usually assume that the process follows a normal time evolution expressed by the considered model and not another one : the correctness of the model is clearly a major problem and it must be checked before the application of any approach.

A natural aim is to characterize the trajectories followed by the system starting from an initial state. In P-time Event Graphs, it is well-known that a simple forward simulation does not guarantee the correct synchronization of the transitions and often leads to token deaths. A first objective is the determination of possible trajectories without token deaths. The concept of extremal (lowest and greatest, see [22]) trajectories is relevant for the class of P-time Event Graphs and corresponds to the earliest and latest trajectories. In this part, the objective is to express the *relations* of these extremal trajectories from the model and the initial condition, for a given horizon.

An important notion is the consistency, which can be defined by the existence of a time trajectory following the model. A second question is to know if the different tasks can be sufficiently repeated during a period such that a given production plan can be performed. More precisely, our objective is to know if the different tasks can be repeated infinitely or during a finite period, and to determine the *maximal horizon* (maximal number of events) where the synchronizations of the transitions can be made.

A consequence is that the end of this horizon is also the limit of consistency which leads to a non-synchronization of the transitions : at least, a token death happens. The last objective is the determination of the *date of the first token deaths*.

In (max, +) algebra, other studies naturally analyse the trajectories. Using a fixed point approach, [15] considers the control of Timed Event Graphs with specifications defined by an interval model. Analysis of the consistency of interval descriptor systems as Time Stream Event Graphs is made by using the spectral vector for a given horizon while the greatest state and control trajectories are numerically calculated by an algorithm. In this paper, in-depth analysis of P-time event graphs is performed and algebraic expressions of extremal trajectories are derived. Polynomial algorithms are proposed for the determination of the maximal horizon of temporal consistency and the calculation of the first date of token deaths. This improves the pseudo-polynomial algorithm of [15] for similar problems.

Another possible approach is to rewrite the system in the form of a polyhedron in conventional algebra [6]. A priori, an application of the algorithms of linear programming can check the existence of an arbitrary trajectory. But, recall that the best algorithms of linear programming are polynomial *in the weak sense*. Contrary to these generic algorithms, we propose here algorithms specific to the considered problem whose complexity is polynomial *in the strong sense*.

If we only consider the problem of consistency, a possible technique is the model-checking which is an enumerative method based on the construction of a state class graph and its analysis.

Some authors [4] apply this approach to T-Time Petri nets where each state class is defined by its marking and a set of firing times of the transitions. Starting from a given class, the firing of each enabled transition generates another class and a procedure establishes the list of the different classes and its connections. Generally speaking, model checking faces a combinatorial blow up of the state-space, commonly known as the state explosion problem, even for small systems [24] : the elementary event graph of the example of Figure 4 in [4] which is composed of two places and two transitions, illustrates this fact. As a consequence, these approaches generally consider a class of models where the graph is finite, that is, the Time Petri Nets are *bounded* and the bounds of the time intervals are defined in the *rational numbers*. In this paper, these assumptions are not taken as the considered models are non-bounded Event Graphs where the values of the temporisations are defined in $\mathbb{R}^+$. Contrary to the polyhedra of the classes which generally are approximations of the possible dates [5], the spaces considered in this paper are exact because the concept of lattice is relevant in the event graphs.

In this section, no hypothesis is made on the structure of the Event Graphs which do not need to be strongly connected. The initial marking must only satisfy the classical condition of liveness (no circuit without token), and the usual hypothesis First In First Out (FIFO) for tokens is made.

2.2 Preliminaries

A monoid is a couple $(S, \oplus)$ where the operation $\oplus$ is associative and presents neutral element ε. A semi-ring S is a triplet $(S, \oplus, \otimes)$ where $(S, \oplus)$ and $(S, \otimes)$ are monoids, $\oplus$ is commutative, $\otimes$ is distributive in relation to $\oplus$ and the zero element ε of $\oplus$ is the absorbing element of $\otimes$ $(\varepsilon \otimes a = a \otimes \varepsilon = \varepsilon)$. A dioid D is an idempotent semi-ring (the operation $\oplus$ is idempotent, that is $a \oplus a = a$). The set $\mathbb{R} \cup \{-\infty\}$, provided with the maximum operation denoted $\oplus$ and the addition denoted $\otimes$ is an example of dioid denoted $\mathbb{R}_{\max} = (\mathbb{R} \cup \{-\infty\}, \oplus, \otimes)$. The neutral elements of $\oplus$ and $\otimes$ are represented by $\varepsilon = -\infty$ and $e = 0$, respectively. The absorbing element of $\otimes$ is ε. Isomorphic to the previous one by bijection : $x \longmapsto -x$, another dioid is $\mathbb{R} \cup \{+\infty\}$, provided with the minimum operation denoted $\wedge$ and the addition denoted $\odot$. The neutral elements of $\wedge$ and $\odot$ are represented by $T = +\infty$ and $e = 0$ respectively. The absorbing element of $\odot$ is ε. The following conventions are made : $T \otimes \varepsilon = \varepsilon$ and $T \odot \varepsilon = T$. The expressions $a \otimes b$ and $a \odot b$ are identical if at least either a or b is a finite scalar. The partial order denoted $\leqslant$ is defined as follows : $x \leqslant y \iff x \oplus y = y \iff x \wedge y = x \iff x_i \leqslant y_i$, for i from 1 to n in $\mathbb{R}^n$. Notation $x < y$ means that $x \leqslant y$ and $x \neq y$. A dioid D is complete if it is closed for infinite sums, and the distributivity of the multiplication with respect to addition applies to infinite sums : $(\forall\ c \in D\ )$ $(\forall\ A \subseteq D)\ c \otimes (\bigoplus_{x \in A} x) = \bigoplus_{x \in A} c \otimes x$. For example, $\overline{\mathbb{R}}_{max} = (\mathbb{R} \cup \{-\infty\} \cup \{+\infty\}, \oplus, \otimes)$ is complete. The set of $n.n$ matrices with entries in the complete dioid D including the two operations $\oplus$ and $\otimes$ is also a complete dioid, which is denoted $D^{n.n}$. The elements of the matrices in the (max, +)

expressions (respectively (min, $+$) expressions) are either finite or ε (respectively T). We can deal with nonsquare matrices if we complete them with rows or columns provided the entries equal ε (respectively T). The different operations obey the usual rules of algebra : the notation $\odot$ refers to the multiplication of two matrices in which the $\wedge-$operation is used instead of $\oplus$. The mapping f is said to be residuated if for all $y \in D$, the least upper bound of subset $\{x \in D \mid f(x) \leq y\}$ exists and lies in this subset. Mapping $x \in (\overline{\mathbb{R}}_{max})^n \mapsto A \otimes x$, defined over $\overline{\mathbb{R}}_{max}$ is residuated (see [3]) and the left $\otimes-$residuation of B by A is denoted by : $A \backslash B = \max\{x \in (\overline{\mathbb{R}}_{max})^n$ such that $A \otimes x \leqslant B\}$; moreover, $A \setminus B = (-A)^t \odot B$ or $A \odot B = (-A)^t \setminus B$ (see the proof of theorem 3.21 in part 3.2.3.2 of [3]) with convention $-\infty - (-\infty) = +\infty$ and $+\infty - (+\infty) = +\infty$.

The Kleene star is defined by : $A^* = \bigoplus_{i=0}^{+\infty} A^i$. Denoted as $G(A)$, an associated graph of square matrix A is deduced from this matrix by associating node i with column i and line i and an arc from node j towards node i with $A_{ij} \neq \varepsilon$. The weight $\mid p \mid_w$ of path p is the sum of the labels (weights) on the edges in the path. The length $\mid p \mid_l$ of path p is the number of edges in the path. A circuit is a path which starts from and ends at the same node. Using the Kleene star, the two following theorems are dual and will be considered in the dioid of matrices.

Theorem 1. *(Theorem 4.75 part 1 in [3]) Consider equation*

$$x = A \otimes x \oplus B \tag{2.1}$$

and inequality

$$x \geq A \otimes x \oplus B \tag{2.2}$$

with A and B in complete dioid D. Then, $A^* \otimes B$ is the least solution of (2.1) and (2.2). ■

Theorem 2. *(Theorem 4.73 part 1 in [3]) Consider equation*

$$x = A \backslash x \wedge B \tag{2.3}$$

and inequality

$$x \leq A \backslash x \wedge B \tag{2.4}$$

with A and B in complete dioid D. Then, $A^* \backslash B$ is the greatest solution of (2.3) and (2.4). ■

For A_{ij} and B_i belonging to $\overline{\mathbb{R}}_{max}$, $A \setminus x \wedge B$ can be written $(-A)^t \odot x \wedge B$.

We shall need the following result in the sequel

Corollary 1. *Corollary $x \geq A \otimes x \oplus B \Longleftrightarrow x \geq A \otimes x$ and $x \geq A^* B$.*
$x \leq A \backslash x \wedge B \Longleftrightarrow x \leq A \backslash x \otimes x$ *and* $x \leq A^* \backslash B$.

Proof. We only consider the first result. The proof is dual for the second part. a) $\Longrightarrow$: it is an application of Theorem 1. b) $\Longleftarrow$: $x \geq A^* B$ implies $x \geq B$. ■

2.3 Models and principle of the approach

In a first part, the definition of P-time Event Graphs is given and the firing interval of each transition is described. Using the dater form, the algebraic model is built. This model will be analyzed in the next parts. An elementary production system is described and the principle of the approach is presented.

2.3.1 P-time Event Graphs

A Petri net is a pair (G, M_0), where $G = (R, V)$ is a bipartite graph with a finite number of nodes (the set V) which are partitioned into the disjoint sets of places P and transitions TR (transitions are denoted x while temporization are denoted T, T^- or T^+) ; R consists of pairs of the form (p_i,x_i) and (x_i,p_i) with $p_i \in P$ and $x_i \in TR$. The initial marking M_0 is a vector of dimension $\mid P \mid$ whose elements denote the number of initial tokens in the respective places. The set ${}^\bullet p$ is the set of input transitions of p and $p^\bullet$ is the set of output transitions of place $p \in P$. The set ${}^\bullet x_i$ (respectively, $x_i^\bullet$) is the set of the input (respectively, output) places of the transition $x_i \in TR$.

For a Petri net with $\mid P \mid$ places and $\mid TR \mid$ transitions, the incidence matrix $W = [W_{ij}]$ is a $\mid P \mid \times \mid TR \mid$ matrix of integers and its entry is given by $W_{ij} = W_{ij}^+ - W_{ij}^-$ where W_{ij}^+ is the weight of the arc from the transition j to the place i and W_{ij}^- is the weight of the arc from the place i to the transition j.

In a Petri net, a firing sequence from a marking M, implies a string of successive markings. The characteristic vector s of a firing sequence S is such that each component is an integer corresponding to the number of firings of the corresponding transition. A marking M reached from initial marking M_0 by firing of a sequence S can be calculated by the fundamental relation : $M = M_0 + W \times s$.

Definition 1. *A Petri net is called an Event Graph if each place has exactly one upstream and one downstream transition.*

P-time Petri nets allow the modeling of discrete event systems with sojourn time constraints of the tokens inside the places. Consistent with the dioid $\overline{\mathbb{R}}_{max}$, we associate a temporal interval defined in $\mathbb{R}^+ \times (\mathbb{R} \cup \{+\infty\})$ with each place. Each place $p_l \in P$ is associated with an interval $[T_l^-, T_l^+]$, where T_l^- is the lower bound and T_l^+ the upper bound. Its initial marking is denoted $(M_0)_l$ or m_l.

Definition 2. *A P-time Event Graph is a triplet (G, M_0, g) where G is an Event Graph, M_0 is the initial marking and the mapping g is defined by $p_l \to [T_l^-, T_l^+]$ with $0 \leq T_l^- \leq T_l^+$ from P to $\mathbb{R}^+ \times (\mathbb{R}^+ \cup \{+\infty\})$.*

The interval $[T_l^-, T_l^+]$ is the static interval of duration time of a token in place p_l. The token must stay in this place during the minimum residence duration T_l^-. Before this duration, the token is in a state of unavailability to fire the outgoing transition. The value T_l^+ is a maximum residence duration after which the token must leave the place p_l (and can contribute to the enabling of the downstream transition). If not, the system falls into a token-dead state. So, the token is available to fire the outgoing transition in the time interval $[T_l^-, T_l^+]$.

A consequence is a possible bad synchronization of each transition which is the outgoing transition of more than one place. This situation occurs when the firing dates of the ingoing transitions of at least two places are incoherent. This non-synchronization can be solved by a prediction of the evolution of the tokens in the places.

Example 1.

Let us consider the P-time Event Graph of Fig. 2.2. The initial marking is $\begin{pmatrix} 1 & 1 & 1 & 1 \end{pmatrix}^t$ and the temporal intervals are : $[a_1, +\infty] = [3, +\infty]$, $[a_2, +\infty] = [6, +\infty]$, $[a_3, b_3] = [1, 2]$ and $[a_4, b_4] = [3, 11]$. Let us consider the following simulation for $k = 0$, 1 and 2.

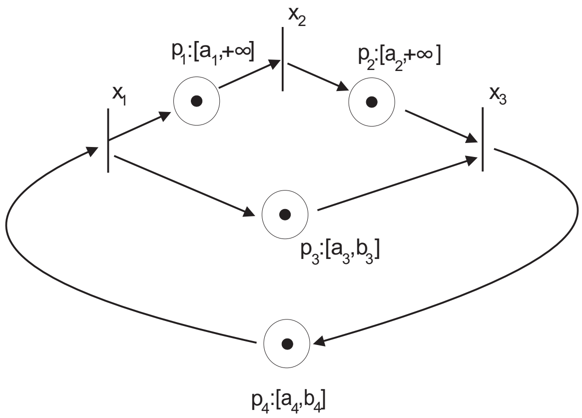

FIG. 2.2 : P-time Event graph

The initial marking is $\begin{pmatrix} 1 & 1 & 1 & 1 \end{pmatrix}^t$ and the temporal intervals are : $[a_1, +\infty] = [3, +\infty]$, $[a_2, +\infty] = [6, +\infty]$, $[a_3, b_3] = [1, 2]$ and $[a_4, b_4] = [3, 11]$. Let us consider the following simulation for $k = 0$, 1 and 2.

k	0	1	2
x_1	4	11	11
x_2	0	7	14
x_3	0	6	13

k	0	1	2
p_1	$[7, +\infty]$	$[14, +\infty]$	$[14, +\infty]$
p_2	$[6, +\infty]$	$[13, +\infty]$	$[20, +\infty]$
p_3	$[5, 6]$	$[12, 13]$	$[12, 13]$
p_4	$[3, 11]$	$[9, 17]$	$[16, 24]$

The first table contains the firing dates while each column k of the second table is the bounds of the sojourn time (in absolute time) of the tokens, produced by the k^{th} firing of the transitions

x_1, x_2 and x_3 in each place. We assume that the tokens of the initial marking are available immediately at $k = 0$. Let us consider the firing of transition x_3 for $k = 3$ which needs to use the tokens present in its upstream places p_2 and p_3 produced at $k = 2$. However, this synchronization does not occur because the interval $[20, +\infty] \cap [12, 13]$ is empty. A consequence is the death of the token in place p_3 at time $t = 13$.

However, transition x_1 can be fired at $t = 18$ because the interval of sojourn time of the token in place p_4 is $[16, 24]$. Therefore, a token is added in place p_3 with time interval $[19, 20]$ and the firing of transition x_3 can occur at time $t = 20$ because the interval $[20, +\infty] \cap [12, 13]$ is replaced by the interval $[20, +\infty] \cap [19, 20]$. ■

Now, we consider the dater form which will give an efficient description.

2.3.2 Dater form

We consider the "dater" description in the (max, +) algebra : each variable $x_i(k)$ represents the date of the k^{th} firing of transition x_i. If we assume a FIFO functioning of the places which guarantees that the tokens do not overtake one another, a correct numbering of the events can be carried out.

In an Event Graph, card($^\bullet p_l$)=card($p_l^\bullet$)=1 for each place $p_l \in P$ and we can associate only a pair (x_i, x_j) with each place $p_l \in P$, such that transition x_j is ingoing ($x_j \in^\bullet p_l$) and transition x_i is outgoing ($x_i \in p_l^\bullet$). Time interval $[a_l, b_l]$ and initial marking m_l are also associated with place p_l. The evolution of the P-time Event Graph is described by the following inequalities expressing relations between the firing dates of transitions :

$\forall p_l \in P$ with $x_j \in^\bullet p_l$ and $x_i \in p_l^\bullet$, $a_l + x_j(k - m_l) \leq x_i(k)$ and $x_i(k) \leq b_l + x_j(k - m_l)$

Now, let us consider a pair of transitions (x_i, x_j) and a given marking q. These conditions define a set of places $P_{i,j,q}$ which can be empty or contain more than one place : Each place p_l of $P_{i,j,q}$ satisfies $\{x_j\} =^\bullet p_l$, $\{x_i\} = p_l^\bullet$ and $m_l = q$. $\forall p_l \in P_{i,j,q}$, we can take the maximum of lower bounds a_l and the minimum of upper bounds b_l and we denote the corresponding values $a^-_{i,j,q} \in \mathbb{R}^+$ and $a^+_{i,j,q} \in \mathbb{R}^-$. More formally, $a^-_{i,j,q} = \bigoplus_{\forall p_l \in P_{i,j,q}} a_l$ and $a^+_{i,j,q} = \bigwedge_{\forall p_l \in P_{i,j,q}} b_l$.

Remark 1. *Naturally, if for each pair of transitions (x_i, x_j), there is a unique place $p_l \in P$ in the Event Graph, we can simplify the notation and replace $a^-_{i,j,q}$ by $a^-_{i,j}$ and $a^+_{i,j,q}$ by $a^+_{i,j}$. In the figures of the paper, each temporisation is directly indexed with the index l of the relevant place p_l.*

Therefore, the system can be described as follows

$\forall P_{i,j,q} \subset P$, $a^-_{i,j,q} + x_j(k - q) \leq x_i(k)$ and $x_i(k) \leq a^+_{i,j,q} + x_j(k - q)$

After permutation of indexes i and j,

and application of simple transformations, the latter inequality is equivalent to $-a^+_{j,i,q} + x_j(k + q) \leq x_i(k)$

In short,

$a^-_{i,j,q} + x_j(k-q) \leq x_i(k)$ and $-a^+_{j,i,q} + x_j(k+q) \leq x_i(k)$

The system can now be expressed with matrices in (max, +) algebra. This allows the writing of a synthetic description on a horizon defined by the maximal initial marking $m = \bigoplus_{\forall p_k \in P} m_{p_k} = \bigoplus_{\forall P_{i,j,q} \subset P} q$.

$$x(k) \geq \bigoplus_{0 \leq q \leq m} A^-_q \otimes x(k-q) \quad (2.5)$$

$$x(k) \geq \bigoplus_{0 \leq q \leq m} A^+_q \otimes x(k+q) \quad (2.6)$$

with $(A^-_q)_{ij} = a^-_{i,j,q}$ if $a^-_{i,j,q}$ exists in $\mathbb{R}$ or ε otherwise,

$(A^+_q)_{ij} = -a^+_{j,i,q}$ if $a^+_{j,i,q}$ exists in $\mathbb{R}$ or ε otherwise.

For instance, in figure 2.3., $(A^-_0)_{3,1} = a^-_{3,1,0} = a_{p_3} = a_3$ and $(A^+_0)_{1,3} = -a^+_{3,1,0} = -b_{p_3} = -b_3$ for place P_3.

This model completely describes the relevant P-time Event Graph by giving a lower bound of state $x(k)$. This lower bound depends on values $x(k-q)$ and $x(k+q)$ for $q = 0$ to m in respectively, inequalities (2.5) and (5.2). As inequality (2.5) corresponds to a classical Timed Event Graph (without assumption of earliest functioning), a P-time Event Graph can be seen as a Timed Event Graph (2.5) following additional specifications (5.2). The Timed Event Graph can express the physical limitations of the process as the minimal cooking time while the upper bounds describe quality criteria on the finished parts and products : The respect of these constraints needs an anticipation of the future behavior of the process. Therefore, the calculation of the lower bound trajectory cannot be made from only the past trajectory like a Timed Event Graph working in the earliest functioning, but must use a prediction of the future evolution. In the sequel, we will see that this remark also holds for the upper bound.

Remark 2. *Some authors add the additional assumption of earliest behavior and replace the inequality in (2.5) by an equality. Therefore, they limit the possibility of modeling of P-time Event Graphs which does not describe a unique trajectory but a set of trajectories. Particularly, P-time Event Graphs can describe uncertainties on sequence time of the process [3] [7] [17] while the minimal and maximal times of each task are exactly known. For instance, the choice of a cooking time in the middle of interval [T_{min},T_{max}] guarantees the quality of the final product but other choices are possible. The choices T_{min} and T_{max} are risky (underdone, overdone) as the parameters of the oven can change.*

The aim of the paper is the analysis of the implicit model defined by (2.5) and (5.2). Before considering the general model, we first introduce the principle of the general approach with a simple example. This part only uses usual algebra. A more general study will be given in the sequel.

2.3.3 Principle of the approach (example 2)

Let us consider an elementary production system composed of two lines in parallel which start at the same time. The process is described by a P-time Event Graph in figure 2.3. The first

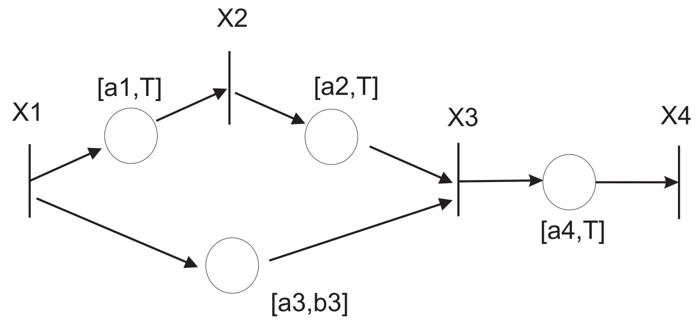

FIG. 2.3 : Principle

line is composed of two tasks while the second one only corresponds to the cooking of a product (a_3). The tasks of line 1 are successively the making of a packet (a_1) and its moving (a_2). When the activities are completed, the finished product is packed (a_4). Naturally, the cooking time b_3 must not be too excessive, otherwise, the product will be damaged ($x_3 > b_3 + x_1$).

The following inequalities describe the two lines. $a_1 + x_1 \leq x_2$, $a_2 + x_2 \leq x_3$ and $a_3 + x_1 \leq x_3 \leq b_3 + x_1$.

Therefore, $a_1 + a_2 + x_1 \leq x_3$ for line 1, and $a_3 + x_1 \leq x_3 \leq b_3 + x_1$ for line 2. So, $a_1 + a_2 + x_1 \leq x_3 \leq b_3 + x_1$ and consequently, condition $a_1 + a_2 \leq b_3$ is necessary otherwise, the system is inconsistent. In other words, the process does not work if time b_3 is less than the sum of the temporizations a_1 and a_2.

Another explanation is as follows. Inequality $x_3 \leq b_3 + x_1$ can be written $-b_3 + x_3 \leq x_1$. From $a_1 + x_1 \leq x_2$; $a_2 + x_2 \leq x_3$; $-b_3 + x_3 \leq x_1$, we can deduce that $-b_3 + a_2 + a_1 + x_1 \leq x_1$. A necessary and sufficient condition of existence of a solution is $-b_3 + a_2 + a_1 \leq 0$.

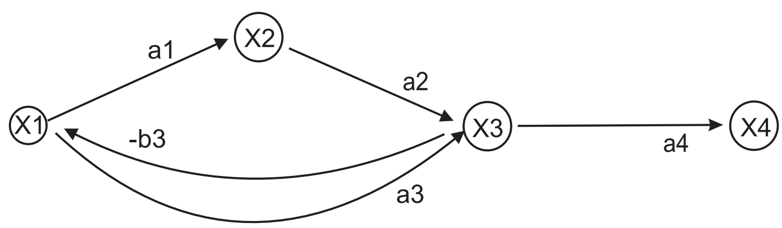

FIG. 2.4 : Associated graph 1

These inequalities can be described by a graph (see Figure 2.4.) defined as follows. The vertices correspond to the transitions of the Petri net and a directed arc from j to i is associated with each inequality $a + x_j \leq x_i$.

This graph shows that term $-b_3 + a_2 + a_1$ is the weight of the circuit defined by transitions x_1, x_2, x_3 and x_1. We can say that the system will be consistent if any circuit of the graph has a negative or null weight. In this case, a sequence of firing dates meeting the consistent system can be found. If the process follows these dates, the production will be satisfactory as each cooked product never stays in oven after delay b_3.

Now, we generalize this first intuitive study and consider the case where the initial marking is null.

2.4 Analysis in the "static" case

Let us assume that the process is static or, in other words that the marking is null : $m = 0$. Therefore, the model described by (2.5) and (5.2) is reduced to the following form.

$$x \geq (A_0^- \oplus A_0^+) \otimes x \tag{2.7}$$

Inequalities of this form are classical in the (max, +) context. The following well-known result clearly shows that the consistency analysis of (2.7) needs an analysis of the circuits in the static case.

Proposition 1. *There is a finite vector $x \in \mathbb{R}^{dim(x)}$ satisfying (2.7) if and only if the associated graph of matrix $A_0^- \oplus A_0^+$ has only circuits with only non-positive weight.*

Recall that $(A^*)_{i,i}$ is the greatest weight of the circuits going by vertex i of the associated graph of matrix A. Another formulation of the proposition is that a necessary and sufficient condition for the existence of a state in $\mathbb{R}$ (not in $\mathbb{R}_{\max}$) is that $((A_0^- \oplus A_0^+)^*)_{i,i}$ converges on $\mathbb{R}_{\max}$ and not on $T = +\infty$ for any index i.

The following form makes the connection with the study in [27]. First, let us note that $x \geq A_0^+ \otimes x$ is equivalent to $x \leq A_0^+ \backslash x$. So, inequality $x \geq (A_0^- \oplus A_0^+) \otimes x$ is equivalent to $A_0^- \otimes x \leq x \leq A_0^+ \backslash x$. This system implies the following expression

$$A_0^- \otimes x \leq A_0^+ \backslash x = (-A_0^+)^t \odot x \tag{2.8}$$

which has been analyzed in the proposition below. It is given with a slightly modified notation.

Proposition 2. *[27]*

There is a finite vector $x \in \mathbb{R}^{dim(x)}$ satisfying $A_0^- \otimes x \leq (-A_0^+)^t \odot x$ if and only if the associated graph of $A_0^+ \otimes A_0^-$ contains circuits with only non-positive weight.

The relation defined by (2.8) has been deduced from (2.7) or, in other words, the set defined by (2.8) includes the set defined by (2.7) but the relations are not mathematically equivalent as shown in the following counter-example.

Example $A_0^- = \begin{pmatrix} -10 & -20 \\ -15 & 0 \end{pmatrix}$ and $(-A_0^+)^t = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$

$$x = \begin{pmatrix} 11 \\ 1 \end{pmatrix}$$

$$A_0^- \otimes x = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \leq (-A_0^+)^t \odot x = \begin{pmatrix} 2 \\ 2 \end{pmatrix} \text{ but } \begin{pmatrix} 1 \\ 1 \end{pmatrix} \leq \begin{pmatrix} 11 \\ 1 \end{pmatrix} \nleq \begin{pmatrix} 2 \\ 2 \end{pmatrix}$$

The following results allow a comparison of the consistency of (2.7) and (2.8) based on the circuits of the associated graphs. The usual multiplication is denoted by a dot below.

Proposition 3. $(\forall \mathbf{k} \in \mathbb{N})(A \oplus B)^{2.k} \geq (A \otimes B)^k$

Proof

The inductive proof is as follows. The hypothesis is H_k :$(A \oplus B)^{2.k} \geq (A \otimes B)^k$.

Initial Step. H_1 defined by $(A \oplus B)^2 \geq (A \otimes B)^1$ is true as $(A \oplus B)^2 = A^2 \oplus A \otimes B \oplus B \otimes A \oplus B^2$

Inductive Step. Let us assume that H_k is true for a given k$\in \mathbb{N}$. We must prove the formula is true for k+1.

$(A \oplus B)^{2.(k+1)} = (A \oplus B)^{2.k} \otimes (A \oplus B)^2 \geq (A \otimes B)^k \otimes (A \otimes B)^1 = (A \otimes B)^{k+1}$ and H_{k+1} is proved. ■

Proposition 4. $(A \oplus B)^* \geq (A \otimes B)^*$

Proof

By definition, $(A \oplus B)^* = \bigoplus_{i \in \mathbb{N}} (A \oplus B)^i \geq \bigoplus_{k \in \mathbb{N}} (A \oplus B)^{2.k}$.

The previous proposition implies that $\bigoplus_{k \in \mathbb{N}} (A \oplus B)^{2.k} \geq \bigoplus_{k \in \mathbb{N}} (A \otimes B)^k = (A \otimes B)^*$ and the proposition is proved. ■

Therefore, the application of the previous result gives :

$(A_0^- \oplus A_0^+)^* \geq (A_0^- \otimes A_0^+)^* \oplus (A_0^+ \otimes A_0^-)^*$. Particularly,

$((A_0^- \oplus A_0^+)^*)_{i,i} \geq ((A_0^- \otimes A_0^+)^*)_{i,i} \oplus ((A_0^+ \otimes A_0^-)^*)_{i,i}$

Consequently, even if $\forall i \in ((A_0^+ \otimes A_0^-)^*)_{i,i} \leq 0$, term $((A_0^- \oplus A_0^+)^*)_{i,i}$ can be positive. Therefore, if the associated graph of $(A_0^+ \otimes A_0^-)$ contains circuits with only non-positive weight, the associated graph of matrix $(A_0^- \oplus A_0^+)$ can have circuits with positive weights. Therefore, application of the previous two propositions shows that inequality (2.7) can be inconsistent while inequality (2.8) is consistent.

In conclusion, this part shows that the consistency depends on the circuits in an associated graph. This analysis will be now generalized to an arbitrary initial marking in the sequel. The implicit model described by (2.5) and (5.2) will first be rewritten on a short horizon in order to simplify the analysis. Then this new form will be used to calculate extremal trajectories and to analyze the consistency in the following sections.

2.5 "Dynamical" model

Now, we consider an arbitrary initial marking. The following proposition is about the existence of a state trajectory in $\mathbb{R}$ (and not in $\mathbb{R}_{\max}$).

Proposition 5. *A necessary condition for the existence of a state trajectory in $\mathbb{R}$ is that the associated graph of matrix $A_0^- \oplus A_0^+$ has only circuits with only non-positive weight.*

Proof. From inequalities (2.5) and (5.2) of the model, we have
$x(k) \geq \bigoplus_{0\leq q\leq m} A_q^- \otimes x(k-q) \oplus \bigoplus_{0\leq q\leq m} A_q^+ \otimes x(k+q) =$

$$(A_0^- \oplus A_0^+) \otimes x(k) \oplus \bigoplus_{1\leq q\leq m} A_q^- \otimes x(k-q) \oplus \bigoplus_{1\leq q\leq m} A_q^+ \otimes x(k+q) \tag{2.9}$$

We can deduce that $x(k) \geq (A_0^- \oplus A_0^+) \otimes x(k)$ and apply the first Proposition in part 1. ■

From (2.9), we deduce the following inequalities, where the right hand term of the first inequality represents the least solution of (2.9) .

$$\begin{cases} x(k) \geq (A_0^- \oplus A_0^+)^* \otimes [\bigoplus_{1\leq q\leq M} A_q^- \otimes x(k-q) \oplus \bigoplus_{1\leq q\leq M} A_q^+ \otimes x(k+q)] \\ x(k) \geq (A_0^- \oplus A_0^+) \otimes x(k) \end{cases} \tag{2.10}$$

The following property shows that (2.10) completely expresses the model.

Proposition 6. ***Proposition.*** *The inequalities (2.10) and the implicit model defined by (2.5) and (5.2) are equivalent.*

Proof : Immediate from Corollary 1.■

Now, let us introduce the following notations.
$$\begin{cases} \mathbb{A}_0^= = A_0^- \oplus A_0^+ \\ \mathbb{A}_q^- = (A_0^- \oplus A_0^+)^* \otimes A_q^- \text{ , for } q = 1 \text{ to } M. \\ \mathbb{A}_q^+ = (A_0^- \oplus A_0^+)^* \otimes A_q^+ \text{ , for } q = 1 \text{ to } M. \end{cases}$$
Therefore, the model (2.10) can be rewritten as follows.

$$\begin{cases} x(k) \geq \mathbb{A}_0^= \otimes x(k) \\ x \geq \bigoplus_{1\leq q\leq M} \mathbb{A}_q^- \otimes x(k-q) \oplus \bigoplus_{1\leq q\leq M} \mathbb{A}_q^+ \otimes x(k+q) \end{cases} \tag{2.11}$$

System (2.11) can be simplified by defining an augmented state vector. We propose below a possible technique. The new state vector denoted $\mathcal{X}$, includes variables $x(k)$, $x_i^-(k)$ and $x_i^+(k)$, for $i = 1$ to $M-1$ defined as follows.

$x_1^-(k) = x(k-1)$, $x_1^+(k) = x(k+1)$,

$x_i^-(k) = x_{i-1}^-(k-1)$ and $x_i^+(k) = x_{i-1}^+(k+1)$, for $i = 2$ to $M-1$.

$\mathcal{X} = \begin{pmatrix} (x^-_{M-1})^t & \dots & (x^-_2)^t & (x^-_1)^t & (x)^t & (x^+_1)^t & (x^+_2)^t & \dots & (x^+_{M-1})^t \end{pmatrix}^t$ (t : transpose). The dimension of $\mathcal{X}$ is denoted n which is equal to the product of the dimension of x by $2.M-1$. The following inequalities completely describe both static part and dynamic part of the system.

$x^-_1(k) = x(k-1) \Leftrightarrow x^-_1(k) \geq x(k-1)$ and $x(k) \geq x^-_1(k+1)$

$x^+_1(k) = x(k+1) \Leftrightarrow x^+_1(k) \geq x(k+1)$ and $x(k) \geq x^+_1(k-1)$

$x^-_i(k) = x^-_{i-1}(k-1) \Leftrightarrow x^-_i(k) \geq x^-_{i-1}(k-1)$ and $x^-_{i-1}(k) \geq x^-_i(k+1)$, for $i = 2$ to $M-1$.

$x^+_i(k) = x^+_{i-1}(k+1) \Leftrightarrow x^+_i(k) \geq x^+_{i-1}(k+1)$ and $x^+_{i-1}(k) \geq x^+_i(k-1)$, for $i = 2$ to $M-1$.

Finally, the simplified inequalities are as follows.

$$\begin{cases} \mathcal{X}(k) \geq \mathcal{A}^= \otimes \mathcal{X}(k) \\ \mathcal{X}(k) \geq \mathcal{A}^- \otimes \mathcal{X}(k-1) \\ \mathcal{X}(k) \geq \mathcal{A}^+ \otimes \mathcal{X}(k+1) \end{cases} \tag{2.12}$$

or

$$\begin{pmatrix} \mathcal{X}(k) \\ \mathcal{X}(k+1) \end{pmatrix} \geq \begin{pmatrix} \mathcal{A}^= & \mathcal{A}^+ \\ \mathcal{A}^- & \mathcal{A}^= \end{pmatrix} \otimes \begin{pmatrix} \mathcal{X}(k) \\ \mathcal{X}(k+1) \end{pmatrix} \tag{2.13}$$

We shall now illustrate the procedure for the synthesis of matrices $\mathcal{A}^=$, $\mathcal{A}^-$ and $\mathcal{A}^+$ in inequalities (2.12). Let $M = 3$. So, $\mathcal{X} = \begin{pmatrix} (x^-_2)^t & (x^-_1)^t & (x)^t & (x^+_1)^t & (x^+_2)^t \end{pmatrix}^t$

$x(k) \geq \mathbb{A}^-_3 \otimes x^-_2(k-1) \oplus \mathbb{A}^-_2 \otimes x^-_1(k-1) \oplus \mathbb{A}^-_1 \otimes x(k-1) \oplus (\mathbb{A}^-_0 \oplus \mathbb{A}^+_0) \otimes x(k)$

$\oplus \mathbb{A}^+_1 \otimes x(k+1) \oplus \mathbb{A}^+_2 \otimes x^+_1(k+1) \oplus \mathbb{A}^+_3 \otimes x^+_2(k+1)$

$x^-_1(k) = x(k-1) \Leftrightarrow x^-_1(k) \geq x(k-1)$ and $x(k) \geq x^-_1(k+1)$

$x^+_1(k) = x(k+1) \Leftrightarrow x^+_1(k) \geq x(k+1)$ and $x(k) \geq x^+_1(k-1)$

$x^-_2(k) = x^-_1(k-1) \Leftrightarrow x^-_2(k) \geq x^-_1(k-1)$ and $x^-_1(k) \geq x^-_2(k+1)$

$x^+_2(k) = x^+_1(k+1) \Leftrightarrow x^+_2(k) \geq x^+_1(k+1)$ and $x^+_1(k) \geq x^+_2(k-1)$

$$\mathcal{A}^= = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \mathbb{A}^=_0 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}, \mathcal{A}^- = \begin{pmatrix} \varepsilon & I & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & I & \varepsilon & \varepsilon \\ \mathbb{A}^-_3 & \mathbb{A}^-_2 & \mathbb{A}^-_1 & I & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & I \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}, \text{and } \mathcal{A}^+ = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ I & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & I & \mathbb{A}^+_1 & \mathbb{A}^+_2 & \mathbb{A}^+_3 \\ \varepsilon & \varepsilon & I & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & I & \varepsilon \end{pmatrix}$$

Also, the diagonal of $\mathcal{A}^-$ can be modified such that the nondecrease of state trajectory is guaranteed. This operation keeps expressions $x^-_1(k) = x(k-1)$, $x^-_2(k) = x^-_1(k-1)$, ... unchanged, because $x^-_2(k) \geq x^-_1(k-1) \oplus x^-_2(k-1) = x^-_1(k-1) \oplus x^-_1(k-2) = x^-_1(k-1)$, for instance.

These expressions describe the "lower" constraints on $\mathcal{X}$ produced by the model which can maximize the state estimation. The set of inequalities (2.12) clearly describes a forward part ($\mathcal{X}(k) \geq \mathcal{A}^- \otimes \mathcal{X}(k-1)$, a backward part and a static (i.e neither backward, nor forward) part ($\mathcal{X}(k) \geq \mathcal{A}^= \otimes \mathcal{X}(k)$). These relations lead to complex backward/forward interconnections which can produce inconsistencies in the model.

Symmetrically, as mapping $\mathcal{A}^{=} \otimes \mathcal{X}(k)$, $\mathcal{A}^{-} \otimes \mathcal{X}(k-1)$ and $\mathcal{A}^{+} \otimes \mathcal{X}(k+1)$ are residuated, the application of property f3 in [3] part 4.4.4) gives the following form : it expresses every "upper" constraint on $\mathcal{X}(k)$ which can minimize it.

$$\begin{cases} \mathcal{X}(k) \leq \mathcal{A}^{=} \backslash \mathcal{X}(k) \\ \mathcal{X}(k) \leq \mathcal{A}^{-} \backslash \mathcal{X}(k+1) \\ \mathcal{X}(k) \leq \mathcal{A}^{+} \backslash \mathcal{X}(k-1) \end{cases} \tag{2.14}$$

Each model can be deduced from the other one by duality and each lower (upper) matrix respectively corresponds to an upper (lower) matrix with the same notation : symbols $\geq$, $\oplus$ and $\otimes$, respectively correspond to $\leq$, $\wedge$ and $\backslash$; Number of events $k-1$ is replaced by $k+1$ and conversely.

In the following part, the time evolution of the model (2.12) is analyzed.

2.6 Extremal acceptable trajectories by series of matrices

Unlike the class of Timed Event Graphs which define a unique trajectory on assumption of earliest behavior, P-time Event Graphs define a set of trajectories which depend on matrices $\mathcal{A}^{=}$, $\mathcal{A}^{-}$ and $\mathcal{A}^{+}$. The aim of this section is the determination of the lowest (respectively, greatest) acceptable trajectories satisfying an initial condition given by $\mathcal{X}(0) \in [\mathcal{X}_0^-, \mathcal{X}_0^+]$. In the sequel, we will show that the existence of a trajectory depends on special new matrices denoted w_k.

As a finite horizon $h \in \mathbb{N}$ is considered, a realistic assumption is that the model operates on the same horizon : we consider the model (2.13) for $k = 0, ..., h-1$.

2.6.1 Lowest state trajectory

The following algorithms give the lowest and greatest trajectory satisfying the objective. The first step a) is the forward calculation of parameters w_k which only depend on the model. Starting from the initial condition $\mathcal{X}_0^-$ (resp. $\mathcal{X}_0^+$), the second step b) is also based on a forward iteration. It expresses a first estimate of the lowest (resp. greatest) trajectory denoted β_k^- (resp. β_k^+), which is finally improved by a maximisation (respectively, a minimisation) in step c). The final result is the lowest (resp. greatest) trajectory denoted by $\mathcal{X}_k^-$ (resp. $\mathcal{X}_k^+$).

Theorem 3. *If the process operates on horizon $h \in \mathbb{N}$ and if matrices w_k defined below have no positive circuit, the lowest state trajectory in $\mathbb{R} \cup \{-\infty\}$ satisfying $\mathcal{X}(0) \geq \mathcal{X}_0^- \in (\mathbb{R} \cup \{-\infty\})^n$ is given by the following forward/backward algorithm.*

Forward/backward algorithm

a) Coefficients of w_k by forward iteration

Initialization : $w_0 = \mathcal{A}^{=}$

for $k = 1$ *to* h, $w_k = \mathcal{A}^= \oplus \mathcal{A}^- \otimes (w_{k-1})^* \otimes A^+$

b) First estimate β_k^- *by forward iteration*

Initialization : $\beta_0^- = \mathcal{X}_0^-$

for $k = 1$ *to* h, $\beta_k^- = \mathcal{A}^- \otimes (w_{k-1})^* \otimes \beta_{k-1}^-$,

c) Trajectory $\mathcal{X}_k^-$ *by backward iteration*

Initialization : $\mathcal{X}_h^- = (w_h)^* \otimes \beta_h^-$

for $k = h-1$ *to* 0, $\mathcal{X}_k^- = (w_k)^* \otimes [\mathcal{A}^+ \otimes \mathcal{X}_{k+1}^- \oplus \beta_k^-]$

Proof Theorem 1 shows that the smallest solution satisfying $\mathcal{X} \geq (\gamma^0\mathcal{A}^= \oplus \gamma^1.\mathcal{A}^- \oplus \gamma^{-1}.\mathcal{A}^+) \otimes \mathcal{X}$ with $\mathcal{X}(0) \geq \mathcal{X}_0^-$ also satisfies the corresponding equality. These can be written by the following equations.

$$\begin{cases} \mathcal{X}(0) = \mathcal{A}^= \otimes \mathcal{X}(0) \oplus \mathcal{A}^+ \otimes \mathcal{X}(1) \oplus \mathcal{X}_0^- \\ \mathcal{X}(k) = \mathcal{A}^= \otimes \mathcal{X}(k) \ \oplus \mathcal{A}^- \otimes \mathcal{X}(k-1) \oplus \\ \mathcal{A}^+ \otimes \mathcal{X}(k+1) \text{ for } k = 1 \text{ to } h-1 \\ \mathcal{X}(h) = \mathcal{A}^= \otimes \mathcal{X}(h) \oplus \mathcal{A}^- \otimes \mathcal{X}(h-1) \end{cases} \tag{2.15}$$

The following proposition $\mathcal{P}(k)$ is now proved by recursion.

$\mathcal{P}(k) : \mathcal{X}^-(k) = (w_k)^* \otimes [\mathcal{A}^+ \otimes \mathcal{X}^-(k+1) \oplus \beta_k^-]$

Base case : $\mathcal{P}(0)$

From the first equality of (2.15), we can write

$\mathcal{X}(0) = w_0 \otimes \mathcal{X}(0) \oplus \mathcal{A}^+ \otimes \mathcal{X}(1) \oplus \beta_0^-$ where $w_0 = \mathcal{A}^=$ and $\beta_0^- = \mathcal{X}_0^-$. Therefore, $\mathcal{X}(0) = (w_0)^*[\mathcal{A}^+ \otimes \mathcal{X}(1) \oplus \beta_0^-]$, which proves $\mathcal{P}(0)$.

Case : $\mathcal{P}(1)$

From the second equality of (2.15), we can write for $k = 1$

$\mathcal{X}(1) = \mathcal{A}^= \otimes \mathcal{X}(1) \oplus \mathcal{A}^- \otimes \mathcal{X}(0) \oplus \mathcal{A}^+ \otimes \mathcal{X}(2)$

If $\mathcal{P}(0)$ is used,

$\mathcal{X}(1) = \mathcal{A}^= \otimes \mathcal{X}(1) \oplus \mathcal{A}^- \otimes [(w_0)^*[\mathcal{A}^+ \otimes \mathcal{X}(1) \oplus \beta_0^-]] \oplus \mathcal{A}^+ \otimes \mathcal{X}(2)$

The distributivity of $\otimes$ with respect to $\oplus$ leads to

$\mathcal{X}(1) = [\mathcal{A}^= \oplus \mathcal{A}^- \otimes (w_0)^* \otimes \mathcal{A}^+] \otimes \mathcal{X}(1) \oplus \mathcal{A}^- \otimes (w_0)^* \otimes \beta_0^- \oplus \mathcal{A}^+ \otimes \mathcal{X}(2) =$

$w_1 \otimes \mathcal{X}(1) \oplus \beta_1^- \oplus \mathcal{A}^+ \otimes \mathcal{X}(2)$ where $w_1 = \mathcal{A}^= \oplus \mathcal{A}^-(w_0)^*\mathcal{A}^+$ and $\beta_1^- = \mathcal{A}^-(w_0)^* \otimes \beta_0^-$. Therefore,

$\mathcal{X}(1) = (w_1)^* \otimes [\mathcal{A}^+ \otimes \mathcal{X}(2) \oplus \beta_1^-]$ and $\mathcal{P}(1)$ is proved. Now, this approach is generalized for $k = 1$ to $h-1$.

Case : $\mathcal{P}(k)$ for k from 1 to $h-1$.

Let us assume $\mathcal{P}(k-1) : \mathcal{X}(k-1) = (w_{k-1})^* \otimes [\mathcal{A}^+ \otimes \mathcal{X}(k) \oplus \beta_{k-1}^-]$. We will prove that $\mathcal{P}(k-1)$ entails $\mathcal{P}(k)$.

From the second equality of (2.15), we can write

$\mathcal{X}(k) = \mathcal{A}^= \otimes \mathcal{X}(k) \oplus \mathcal{A}^- \otimes \mathcal{X}(k-1) \oplus \mathcal{A}^+ \otimes \mathcal{X}(k+1)$

As $\mathcal{X}(k-1) = (w_{k-1})^* \otimes [\mathcal{A}^+ \otimes \mathcal{X}(k) \oplus \beta_{k-1}^-]$, the expression below is deduced :

$\mathcal{X}(k) = \mathcal{A}^{=} \otimes \mathcal{X}(k) \oplus \mathcal{A}^{-} \otimes (w_{k-1})^{*} \otimes [\mathcal{A}^{+} \otimes \mathcal{X}(k) \oplus \beta_{k-1}^{-}] \oplus \mathcal{A}^{+} \otimes \mathcal{X}(k+1)$

The distributivity of $\otimes$ with respect to $\oplus$ yields

$\mathcal{X}(k) = [\mathcal{A}^{=} \oplus \mathcal{A}^{-} \otimes (w_{k-1})^{*} \otimes \mathcal{A}^{+}] \otimes \mathcal{X}(k) \oplus \mathcal{A}^{-} \otimes (w_{k-1})^{*} \otimes \beta_{k-1}^{-} \oplus \mathcal{A}^{+} \otimes \mathcal{X}(k+1) =$

$w_k \otimes \mathcal{X}(k) \oplus \beta_k^{-} \oplus \mathcal{A}^{+} \otimes \mathcal{X}(k+1)$, where $w_k = \mathcal{A}^{=} \oplus \mathcal{A}^{-} \otimes (w_{k-1})^{*} \otimes \mathcal{A}^{+}$and $\beta_k^{-} = \mathcal{A}^{-} \otimes (w_{k-1})^{*} \otimes \beta_{k-1}^{-}$

Therefore, $\mathcal{X}(k) = (w_k)^{*}[\mathcal{A}^{+} \otimes \mathcal{X}(k+1) \oplus \beta_k^{-}]$ and the desired expression is obtained : $\mathcal{P}(k)$ has been deduced from $\mathcal{P}(k-1)$. Moreover, as $\mathcal{P}(0)$ is true, $\mathcal{P}(k)$ has been proved for k from 1 to $h-1$: the recursion is finished. Knowing β_k^{-}, the calculation of $\mathcal{X}(k)$ uses a backward iteration, while the calculation of β_k^{-} is relevant to a forward iteration.

Now, the final case will be proved.

Case : $\mathcal{P}(h)$

The last equality of (2.15) can be considered like the second equality but without $\mathcal{A}^{+} \otimes \mathcal{X}(k+1)$: the argument of case $\mathcal{P}(k)$ can be taken and we can write

$\mathcal{X}(h) = (w_h)^{*} \otimes \beta_h^{-}$ with $w_h = \mathcal{A}^{=} \oplus \mathcal{A}^{-} \otimes (w_{h-1})^{*} \otimes \mathcal{A}^{+}$and $\beta_h^{-} = \mathcal{A}^{-} \otimes (w_{h-1})^{*} \otimes \beta_{h-1}^{-}$

Finally, as matrices w_k have no positive circuit and $\mathcal{X}_0^{-}$ belongs to $(\mathbb{R} \cup \{-\infty\})^n$, the state trajectory is defined in $\mathbb{R} \cup \{-\infty\}$. ■

2.6.2 Greatest state trajectory

The following theorem can be deduced from the previous one by duality. Steps a) are identical.

Theorem 4. *If the process operates on horizon $h \in \mathbb{N}$ and if matrices w_k defined below have no positive circuit, the greatest state trajectory in $\mathbb{R} \cup \{+\infty\}$ satisfying $\mathcal{X}(0) \leq \mathcal{X}_0^{+} \in (\mathbb{R} \cup \{+\infty\})^n$ is given by the following forward/backward algorithm.*

Forward/backward algorithm

a) Coefficients of w_k by forward iteration

Initialization : $w_0 = \mathcal{A}^{=}$

for $k = 1$ to h, $\quad w_k = \mathcal{A}^{=} \oplus \mathcal{A}^{-} \otimes (w_{k-1})^{*} \otimes \mathcal{A}^{+}$

b) First estimate β_k^{+} by forward iteration

Initialization : $\beta_0^{+} = \mathcal{X}_0^{+}$

for $k = 1$ to h, $\quad \beta_k^{+} = ((w_{k-1})^{*} \otimes \mathcal{A}^{+}) \backslash \beta_{k-1}^{+}$

c) Trajectory $\mathcal{X}_k^{+}$ by backward iteration

$\mathcal{X}_h^{+} = (w_h)^{*} \backslash \beta_h^{+}$

for $k = h-1$ to 0, $\mathcal{X}_k^{+} = (w_k)^{*} \backslash [\mathcal{A}^{-} \backslash \mathcal{X}_{k+1}^{+} \wedge \beta_k^{+}]$

Proof The proof is omitted as it can be deduced by duality from the previous theorem. ■

To sum up, the two algorithms allow the determination of the lowest (respectively, greatest) acceptable trajectories satisfying $\mathcal{X}(0) \geq \mathcal{X}_0^{-}$ (respectively, $\mathcal{X}(0) \leq \mathcal{X}_0^{+}$). They also allow the checking of the existence of a trajectory satisfying $\mathcal{X}(0) \in [\mathcal{X}_0^-, \mathcal{X}_0^+]$ if constraints $\mathcal{X}(0) \leq \mathcal{X}_0^+$ and $\mathcal{X}(0) \geq \mathcal{X}_0^-$ are respectively added in the corresponding algorithms.

Remark 3. *Defined on a box $[\mathcal{X}_0^-, \mathcal{X}_0^+]$, the initial condition is less restrictive than the more usual $\mathcal{X}(0) = \mathcal{X}_0$ which is a particular case. In a natural way, checking this case is made as follows. The determination of the lowest trajectory such as $\mathcal{X}(0) \in [\mathcal{X}_0, \mathcal{X}_0^+]$ with $\mathcal{X}_0 \leq \mathcal{X}_0^+$, allows checking the acceptability of $\mathcal{X}_0$ or in other words, if there is a solution $\mathcal{X}$ so that $\mathcal{X}(0) = \mathcal{X}_0$. Similarly, the determination of the greatest trajectory such as $\mathcal{X}(0) \in [\mathcal{X}_0^-, \ \mathcal{X}_0]$ with $\mathcal{X}_0^- \leq \mathcal{X}_0$ also allows checking the existence of a solution $\mathcal{X}$ so that $\mathcal{X}(0) = \ \mathcal{X}_0$.*

Remark 4. *The calculation of the state trajectories starts from values $(w_h)^* \otimes \beta_h^-$ and $(w_h)^* \backslash \beta_h^+$ and consequently depends on horizon h. The calculation of w_k, β_k^- and β_0^+ depends on index k, but not on horizon h.*

2.7 Consistency

In this paper, an acceptable behavior of the considered P-time Event Graph is defined by any operation guaranteing the liveness of tokens. Therefore, it does not lead to any deadlock situation. As this behavior is represented by the algebraic model (2.12), the aim of this part is to study the existence of a state trajectory solution to these inequalities. Clearly, if we can calculate an arbitrary trajectory starting from box $[\mathcal{X}_0^-, \mathcal{X}_0^+]$, we can deduce that the system is consistent on the considered horizon. We introduce the following notation.

Definition 3. *A dynamic associated graph of square matrices A, B and C on horizon h, denoted by $G_h(A, B, C)$, is deduced from these matrices by associating a node j_k with column j and a node i_k with row i , for $k = 0$ to h. The pattern is as follows : a) An arc from node j_{k-1} towards node i_k if $A_{ij} \neq \varepsilon$; b) An arc from node j_k towards node i_k if $B_{ij} \neq \varepsilon$; c) An arc from node j_k towards node i_{k-1} if $C_{ij} \neq \varepsilon$.*

In this paper, we consider $G_h(\mathcal{A}^-, \mathcal{A}^=, \mathcal{A}^+)$. An example is given in figure 2.5. The dimension of each column k is dimension n of the state. As in the static case presented in section 2.4, the system is consistent if the dynamic associated graph $G_h(\mathcal{A}^-, \mathcal{A}^=, \mathcal{A}^+)$ has no circuit with positive weight. These circuits can simply be situated in the associated graph of the static part ($\mathcal{A}^=$) or of the dynamic part ($\mathcal{A}^-$ and $\mathcal{A}^+$). Figure 2.7 shows that the circuits can present a complex form.

Assuming the liveness of the Event Graph, the following theorem considers the temporal consistency of P-time Event graphs. This theorem is about the existence of a state trajectory in $\mathbb{R}$, and not in $\mathbb{R}_{\max}$.

Theorem 5. *A live P-time Event Graph is consistent on arbitrary horizon h if and only if the dynamic associated graph $G_h(\mathcal{A}^-, \mathcal{A}^=, \mathcal{A}^+)$ contains circuits with only non-positive weight.*

Proof The model can completely be represented by system (2.15) after replacing symbol $=$ with $\geq$. This system can be rewritten in $\mathbb{R}_{\max}$ under the global form $x \geq A \otimes x \oplus B$ which includes every inequality. The relevant dynamic associated graph is $G_h(\mathcal{A}^-, \mathcal{A}^=, \mathcal{A}^+)$. As the least solution is A^*B, this system has at least a solution in $\mathbb{R}_{\max}$ if the global matrix A has no strictly positive circuits. This gives a sufficient condition for the existence of a solution in $\mathbb{R}_{\max}$. Moreover, $\mathcal{X}_0^-$ can be taken finite : as $\mathcal{X}(0) \geq \mathcal{X}_0^-$ and as the trajectory is nondecreasing by construction of $\mathcal{A}^-$, each component of the state trajectory is different from ε and the trajectory belongs to $\mathbb{R}$. Conversely, if finite $\mathcal{X}$ satisfies the model of the P-time Event Graph, it can only satisfy subsystems with non-positive circuits. ■

The following result is immediate.

Corollary 2. *A live P-time Event Graph with a null initial marking, is consistent if and only if the dynamic associated graph of matrix $\mathcal{A}^=$ contains circuits with only non-positive weight.*

Remark 5. *A live P-time Event Graph whose initial marking is null is without circuit (in the Event Graph), but the dynamic associated graph of matrix $\mathcal{A}^=$ can have circuits.*

Now, we consider matrices w_k which allow a characterization of the circuits. The following property gives a graphical interpretation of the calculation of these matrices.

Property 1. *Entry $((w_h)^*)_{i_h,j_h}$ represents the maximum weight of all the paths from vertices j_h to vertices i_h for i, $j \in [1..n]$ in the dynamic associated graph $G_h(\mathcal{A}^-, \mathcal{A}^=, \mathcal{A}^+)$ except the paths containing an arc from index 0 node to index 0 node.*

Proof

Let us consider the relations inside horizon [0, 1]. So, $\mathcal{X}(1)_i \geq (\mathcal{A}^-)_{i,l} \otimes \mathcal{X}(0)_l \geq (\mathcal{A}^-)_{i,l} \otimes (\mathcal{A}^+)_{l,j}\mathcal{X}(1)_j$ but also, $\mathcal{X}(1)_i \geq (\mathcal{A}^=)_{i,j}\mathcal{X}(1)_j$. So, $(w_1)_{i,j} = (\mathcal{A}^= \oplus \mathcal{A}^- \otimes \mathcal{A}^+)_{i,j}$ represents the greatest weight on the following paths :

- an arc $(j_1 \to i_1)$ (matrix $\mathcal{A}^=$)
- or two successive arcs $(j_1 \to l_0)$ and $(l_0 \to i_1)$ (product $\mathcal{A}^- \otimes \mathcal{A}^+$).

Expression $((w_1)^*)_{i,j}$ represents the greatest weight on the following paths (by default, the weight is zero if there is no path between two vertices) going successively through,

- nodes of indexes j_1 to i_1 and again (expressed by $(\mathcal{A}^=)^*$),
- or nodes of indexes j_1 to l_0 and l_0 to i_1 and again (expressed by $(\mathcal{A}^- \otimes \mathcal{A}^+)^*$),
- or nodes of indexes j_1 to l_1, then l_1 to k_0 and k_0 to i_1 (expressed by $(\mathcal{A}^- \otimes \mathcal{A}^+)(\mathcal{A}^=)$),

, and so on.

Remark 6. *As there is no term like $\mathcal{A}^- \otimes \mathcal{A}^= \otimes \mathcal{A}^+$, $(w_1)^*$ is not the result of paths containing an arc from node of index j_0 to node of index i_0 directly.*

For horizon [0, 2], $(w_2)_{ij} = (\mathcal{A}^= \oplus \mathcal{A}^- \otimes (w_1)^* \otimes \mathcal{A}^+)_{ij}$ represents the greatest weight on the following paths : :

- an arc $(j_2 \to i_2)$ (matrix $\mathcal{A}^=$)

- or, an arc $(j_2 \to l_1)$ (matrix $\mathcal{A}^+$), the previous paths from l_1 to m_1 expressed by matrix $(w_1)^*$ (described above) and an arc $(l_1 \to i_2)$ (matrix $\mathcal{A}^-$).

Consequently, $(w_2)^*$ represents the greatest weight of every path (and circuit) of the associated graph on horizon [0, 2] and defined by a path from i_2 to i_2 and going possibly to nodes of indexes 1 and 0, except the paths containing an arc from nodes of indexes 0 to 0.

The argument can be repeated until $k = h$. ■

The following Theorem improves Theorem 5 by giving a practical way to check the consistency.

Theorem 6. *A live P-time Event Graph is consistent if and only if the associated graph of each matrix w_k for any $k \geq 0$ contains circuits with only non-positive weight.*

Proof. Property 1 says that matrices $(w_k)^*$ represent the greatest weight of almost every path and circuit in $G_h(\mathcal{A}^-, \mathcal{A}^=, \mathcal{A}^+)$. The weights of circuits which are not "present" in $(w_k)^*$ are "present" in $(w_{k+1})^*$ for $G_{h+1}(\mathcal{A}^-, \mathcal{A}^=, \mathcal{A}^+)$ as the associated graph is the repetition of a pattern. Consequently, each circuit is expressed and the proof can be deduced from Theorem 5. ■

Therefore, if there is an index k_1 such that an entry $((w_k)^*)_{i,j}$ is infinite, we can conclude that there is a path from j to i, containing a circuit with a positive weight in $G_{k_1}(\mathcal{A}^-, \mathcal{A}^=, \mathcal{A}^+)$. So, the system is not consistent on horizon greater than $h \geq k_1$. An example of circuit with positive circuit is given in figure 2.7.

The following result will facilitate the analysis of the consistency and its checking.

Property 2. $w_k \geq w_{k-1}$ *for* $k \geq 1$

Proof

Let us suppose that $w_{k-1} \geq w_{k-2}$, $w_k = \mathcal{A}^= \oplus \mathcal{A}^- \otimes (w_{k-1})^* \otimes \mathcal{A}^+ =$
$\mathcal{A}^= \oplus \mathcal{A}^- \otimes \mathcal{A}^+ \oplus \mathcal{A}^- \otimes (w_{k-1}) \otimes \mathcal{A}^+ \oplus \mathcal{A}^- \otimes (w_{k-1})^2 \otimes \mathcal{A}^+ \oplus ...$

As $w_{k-1} \geq w_{k-2}$ and by isotony of the product, $w_k \geq \mathcal{A}^= \oplus \mathcal{A}^- \otimes \mathcal{A}^+ \oplus \mathcal{A}^- \otimes (w_{k-2}) \otimes \mathcal{A}^+ \oplus \mathcal{A}^- \otimes (w_{k-2})^2 \otimes \mathcal{A}^+ \oplus ... = w_{k-1}$

Moreover, $w_1 \geq w_0 = \mathcal{A}^=$ as $w_1 = \mathcal{A}^= \oplus \mathcal{A}^- \otimes (w_0)^* \otimes \mathcal{A}^+$. So, the series $w_0 = \mathcal{A}^=$ and $w_k = \mathcal{A}^= \oplus \mathcal{A}^- \otimes (w_{k-1})^* \otimes \mathcal{A}^+$ for $k \geq 1$ is nondecreasing. ■

Suppose that there is an index k_1 such that matrix w_k does not increase ($w_{k_1+1} = w_{k_1}$) with w_k belonging to $\mathbb{R}_{\max}$. From Property 2, we can conclude that no matrix w_k has positive circuit for any index k. Consequently, the P-time Event graph is consistent on an infinite horizon. In this case, the tests show that the convergence of consistent P-time Event Graphs is fast.

2.7.1 Example 3

The following example illustrates the results about consistency and extremal trajectories. Computation tests are made using the max-plus toolbox in Scilab.

2.7.1.1 Model

A slight modification of the example 2 in Figure 2.3 gives the closed-loop structure of the example 1 in Figure 2.2 which describes a limitation of resources. An upper bound on packing (b_4) is added. The initial marking is $\begin{pmatrix} 1 & 1 & 1 & 1 \end{pmatrix}^t$ and $a_1 = 3$, $a_2 = 3$, $a_3 = 1$, $a_4 = 3$, $b_3 = 2$, $b_4 = 11$. It is almost example 1 but with a modification of a_2. Therefore,

$$A_0^- = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \end{pmatrix} A_0^+ = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \end{pmatrix}$$

$\mathbb{A}_0^= = A_0^- \oplus A_0^+ = \varepsilon$

$$A_1^- = \begin{pmatrix} \varepsilon & \varepsilon & 3 \\ 3 & \varepsilon & \varepsilon \\ 1 & 3 & \varepsilon \end{pmatrix} A_1^+ = \begin{pmatrix} \varepsilon & \varepsilon & -2 \\ \varepsilon & \varepsilon & \varepsilon \\ -11 & \varepsilon & \varepsilon \end{pmatrix}$$

Matrices $\mathcal{A}^=$, $\mathcal{A}^-$ and $\mathcal{A}^+$ are now deduced.

$\mathcal{A}^= = \mathbb{A}_0^= = A_0^- \oplus A_0^+ = \varepsilon$, $\mathcal{A}^- = \mathbb{A}_1^- = (\mathbb{A}_0^=)^* \otimes A_1^- = A_1^-$ and $\mathcal{A}^+ = \mathbb{A}_1^+ = (\mathbb{A}_0^=)^* \otimes A_1^+ = A_1^+$ The relevant associated graph is in figure 2.5.

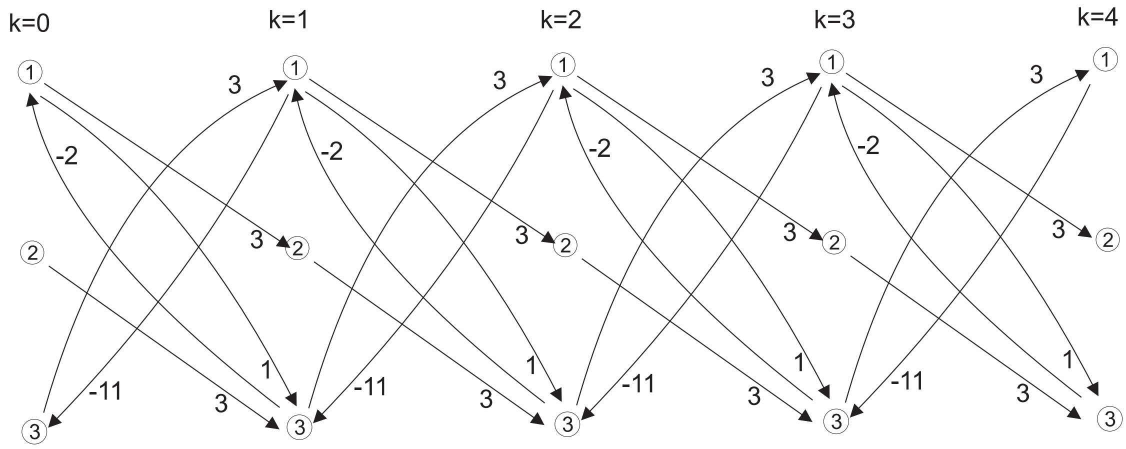

FIG. 2.5 : Associated graph in consistent case

2.7.1.2 Series

$w_0 = \mathcal{A}^= = \varepsilon$

$$w_1 = \begin{pmatrix} -8 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 1 \\ \varepsilon & \varepsilon & -1 \end{pmatrix}, w_2 = \begin{pmatrix} -8 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 1 \\ -7 & \varepsilon & -1 \end{pmatrix}, w_3 = \begin{pmatrix} -8 & \varepsilon & -6 \\ \varepsilon & \varepsilon & 1 \\ -7 & \varepsilon & -1 \end{pmatrix}, w_4 = \begin{pmatrix} -8 & \varepsilon & -6 \\ -14 & \varepsilon & 1 \\ -7 & \varepsilon & -1 \end{pmatrix}$$

The calculation of matrices w_k shows that they are constant after a short transitory period ($w_k = w_4$ for $k \geq 5$) and that they have no positive circuit. Consequently, the system is consistent on an arbitrary horizon.

2.7.1.3 Lowest and greatest state trajectories

Now, we apply the algorithms of Theorems 3 and 4 which provide lowest and greatest state trajectories, χ^+ and χ^+ respectively. The arbitrary initial conditions are $\chi_0^- = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}^t$ and $\chi_0^+ = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}^t$. In step b), intermediate values β^- and β^+, which are given in the following tables, are deduced from matrices w_k and initial conditions χ_0^- and χ_0^+ by a forward approach. They give a first estimate of the lowest and greatest trajectories.

k	0	1	2	3	4	5	6	7	8	9	10
β_1^-	1	3	6	10	14	18	22	26	30	34	38
β_2^-	0	4	6	9	13	17	21	25	29	33	37
β_3^-	0	3	7	11	15	19	23	27	31	35	39

k	0	1	2	3	4	5	6	7	8	9	10
β_1^+	1	11	14	24	27	36	40	49	53	62	66
β_2^+	0	T	T	T	T	T	T	T	T	T	T
β_3^+	0	3	13	16	25	29	38	42	51	55	64

Intermediate values β^- and β^+ are now improved by the backward step c). The following tables contain lowest and greatest state trajectories, χ^- and χ^+ respectively.

k	0	1	2	3	4	5	6	7	8	9	10
$\mathcal{X}_1^-$	1	5	9	13	17	21	25	29	33	37	38
$\mathcal{X}_2^-$	0	4	8	12	16	20	24	28	32	36	40
$\mathcal{X}_3^-$	0	3	7	11	15	19	23	27	31	35	39

k	0	1	2	3	4	5	6	7	8	9	10
$\mathcal{X}_1^+$	1	10	14	23	27	36	40	49	53	62	66
$\mathcal{X}_2^+$	0	9	13	22	26	35	39	48	52	61	T
$\mathcal{X}_3^+$	0	3	12	16	25	29	38	42	51	55	64

The following table is the state trajectory of the Timed Event Graph using the lower bound of the temporisations of the P-time Event Graph (see figure 2.2). With the assumption of earliest behavior,

$x(k) = A \otimes x(k)$ with $A = A_1^-$. As $x(0) = \chi_0^- = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}^t$, a comparison can be made.

k	0	1	2	3	4	5	6	7	8	9	10
x_1	1	3	6	10	12	15	19	21	24	28	30
x_2	0	4	6	9	13	15	18	22	24	27	31
x_3	0	3	7	9	12	16	18	21	25	27	30

Figure 2.6 shows the trajectories of transition x_1 : The lowest and greatest trajectories for the P-time Event Graph ; The trajectory of the relevant Timed Event Graph. The three trajectories are clearly different ($x(k) \leq \chi^-(k) \leq \chi^+(k)$) as their rates (the calculation of the different cycle times is made in [12]).

Remark 7. *It is important to note that each extremal trajectory depends on the lower and upper bounds of the temporisations and not only, on one of those. The calculation of the minimal trajectory naturally requires not only the inequalities corresponding to Timed Event Graphs (2.5) but also the upper constraints (5.2) : the example in figure 2.6 shows that the earliest functioning of a Timed Event Graph using the relevant equality of (2.5) does not satisfy the inequalities of the P-time Event Graph in the autonomous case. This fact entails that the trajectories of P-time Event Graphs cannot always be deduced by a direct forward iteration like in the state equation in Timed Event Graphs. Note that in this example, the P-time Event Graph is consistent.*

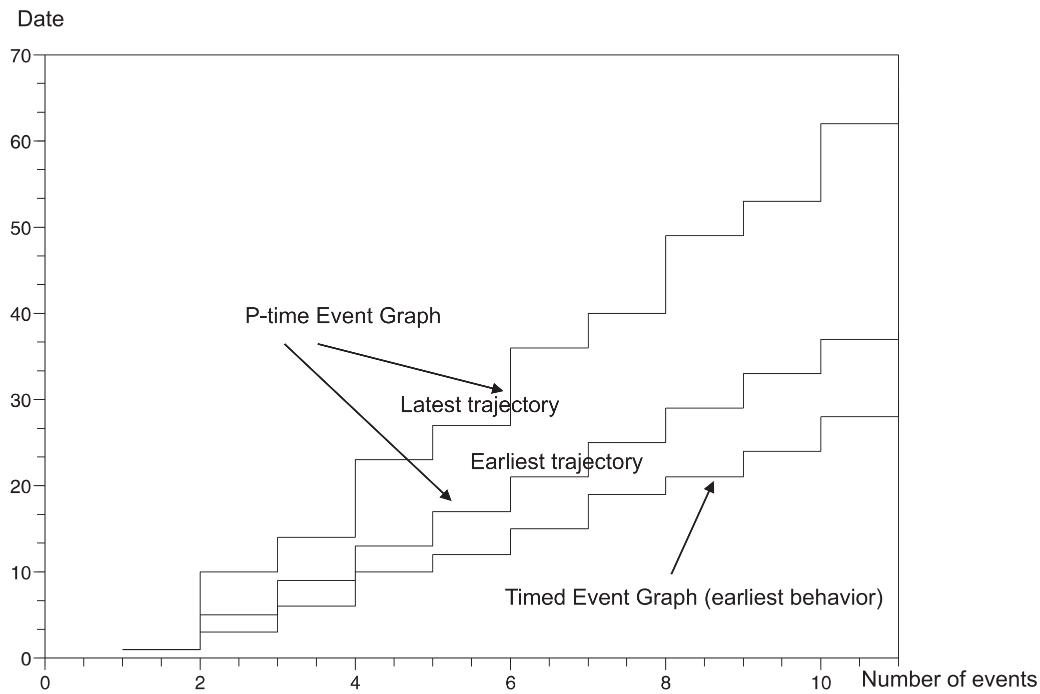

FIG. 2.6 : Trajectories

2.7.2 Maximal horizon of temporal consistency

Assuming $\mathcal{B}$ the liveness of the Event Graph, we consider the temporal consistency of P-time Event graphs. Clearly, if we can calculate an arbitrary finite trajectory (that is, in $\mathbb{R}$) starting from $\mathcal{X}(0) \in (\mathbb{R})^n$, the system is consistent on the given horizon. Therefore, the liveness of tokens is guaranteed and it does not lead to any deadlock situation. In the previous part, we prove that the existence of a finite trajectory only depends on matrices w_k and more precisely, that a finite trajectory exists if and only if matrices $(w_k)^*$ converge in $\mathbb{R}_{\max}$.

Let us now consider the problem of the determination of the maximal horizon of temporal consistency. In step c) of the algorithms, the calculations of the state trajectory $\mathcal{X}^-(k)$ start from values w_h , β_h^- and β_h^+ and consequently depend on horizon $[0, h]$ where h is a datum. Contrary to step c), the calculations of w_k, β_k^- and β_k^+ start from $\mathcal{A}^=$, $\mathcal{X}_0^-$ and $\mathcal{X}_0^+$ in steps a) and b) : they depend on index k, but not on horizon h as the calculations can continue after h. Therefore, the problem is now to determine the maximal horizon h_{max} where the system can follow a finite trajectory. As the horizon can be finite or infinite, we consider the two following cases.

- Case 1. Matrix $(w_k)^*$ does not belong to $\mathbb{R}_{\max}$. As there is at least an infinite entry $((w_k)^*)_{i,j} = +\infty$, the system is not consistent on horizon $[0, h]$ with $h \geq k$.
- Case 2. Matrix $(w_k)^*$ belongs to $\mathbb{R}_{\max}$. If $w_k = w_{k-1}$, then the P-time Event graph is consistent on an infinite horizon and $h_{max} = +\infty$.

A practical way to determine the horizon of consistency h_{max} is as follows.

Algorithm

Initialization : $k \leftarrow 0$. Calculate and analyze $(w_k)^*$ for $k \geq 0$. Stop if case 1 ($h_{max} = k - 1$ if $k \geq 1$) or case 2 ($h_{max} = +\infty$) defined above is satisfied or repeat with $k \leftarrow k + 1$. ■

As the series w_k is non-decreasing, each entry converges to a stable finite value or the infinite value $+\infty$.

Example 1 continued.

Now, assume that a failure appears in the moving of the packet whose normal duration a_2 associated with P_2 is equal to 3 in the example 3, which corresponds to $(\mathcal{A}_1^-)_{3,2}$: the duration is now equal to 6 which corresponds to the example 1. So, $\mathcal{A}^= = \varepsilon$, $\mathcal{A}^- = \begin{pmatrix} \varepsilon & \varepsilon & 3 \\ 3 & \varepsilon & \varepsilon \\ 1 & 6 & \varepsilon \end{pmatrix}$ and $\mathcal{A}^+ = \begin{pmatrix} \varepsilon & \varepsilon & -2 \\ \varepsilon & \varepsilon & \varepsilon \\ -11 & \varepsilon & \varepsilon \end{pmatrix}$. The relevant matrices w_k are as follows. We also give w_3^*.

$$w_0 = \mathcal{A}^= = \varepsilon$$

$$w_1 = \begin{pmatrix} -8 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 1 \\ \varepsilon & \varepsilon & -1 \end{pmatrix}, w_2 = \begin{pmatrix} -8 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 1 \\ -4 & \varepsilon & -1 \end{pmatrix}, w_3 = \begin{pmatrix} -8 & \varepsilon & -3 \\ \varepsilon & \varepsilon & 1 \\ -4 & \varepsilon & 1 \end{pmatrix}, w_3^* = \begin{pmatrix} T & \varepsilon & T \\ T & 0 & T \\ T & \varepsilon & T \end{pmatrix}$$

$, w_4 = \begin{pmatrix} T & \varepsilon & T \\ T & \varepsilon & T \\ T & \varepsilon & T \end{pmatrix}$

As some coefficients of w_3^* and also, w_4 are equal to $T = +\infty$, the system is not consistent. Therefore, even if the underlying graph of the Event Graph is live in the usual sense, it is not consistent for $a_2 = 6$. So, as $(w_3)_{3,3} = 1$, $\chi_3^-(k) \geq 1 \otimes \chi_3^-(k)$ which is inconsistent. This incoherence comes from the following inequalities. Figure 2.7 shows the relevant circuit with positive weight.

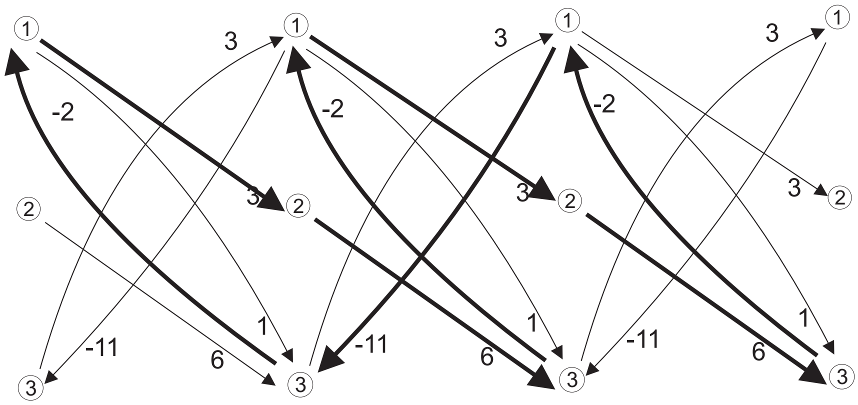

FIG. 2.7 : Circuit with positive weight in inconsistent case

$\chi_3^-(k) \geq 6 \otimes \chi_2^-(k-1)$
$\chi_2^-(k-1) \geq 3 \otimes \chi_1^-(k-2)$
$\chi_1^-(k-2) \geq (-2) \otimes \chi_3^-(k-1)$
$\chi_3^-(k-1) \geq 6 \otimes \chi_2^-(k-2)$
$\chi_2^-(k-2) \geq 3 \otimes \chi_1^-(k-3)$
$\chi_1^-(k-3) \geq (-2) \otimes \chi_3^-(k-2)$
$\chi_3^-(k-2) \geq (-11) \otimes \chi_1^-(k-1)$
$\chi_1^-(k-1) \geq (-2) \otimes \chi_3^-(k)$

As some coefficients of w_3^* are equal to $+\infty$, $h_{max} = 2$ and a trajectory can only be defined on horizon $[0, 2]$. System (5.2) is only consistent for $k = 0$ and 1. The trials show that the tolerance margin of a_2 where the system is consistent, is $[0, 5.5]$. ■

2.7.3 Date of the first token deaths

If the system is only consistent on horizon h_{max}, an admissible trajectory can be calculated but the tokens produced by the firing at date $\mathcal{X}(h_{max})$ do not lead to a complete firing of the transitions at the following number of events $h_{max} + 1$. Below, we consider only the case of places with unitary initial marking : by reason of the lack of space, the case of places with a null initial marking is omitted but follows a similar technique. If $m_l = 1$, the time interval of token stay is $[\mathcal{A}^{-}_{ig} \otimes \mathcal{X}_g(k), \mathcal{A}^{+}_{gi} \backslash \mathcal{X}_g(k]$ for a token generated par the k^{th} firing of transition g in a place $p_l \in P$ such that $\mathcal{X}_g \in^{\bullet} p_l$ and $\mathcal{X}_i \in p_l^{\bullet}$. As there is at least one transition i such that relation $\bigoplus_{j \in^{\bullet}(^{\bullet}\mathcal{X}_i)} \mathcal{A}^{-}_{ij} \otimes \mathcal{X}_j(h_{max}) \leq \mathcal{X}_i(h_{max}+1) \leq \bigwedge_{j \in^{\bullet}(^{\bullet}\mathcal{X}_i)} \mathcal{A}^{+}_{ji} \backslash \mathcal{X}_j(h_{max})$ is not satisfied, the non-synchronization of transition i leads to some token deaths. Let G be the set of transitions $g \in^{\bullet} (^{\bullet}\mathcal{X}_i)$ such that $\mathcal{A}^{+}_{gi} \backslash \mathcal{X}_j(h_{max}) = \bigwedge_{j \in^{\bullet}(^{\bullet}\mathcal{X}_i)} \mathcal{A}^{+}_{ji} \backslash \mathcal{X}_j(h_{max})$. Each input transition $g \in G$ generates a token which dies in the place $p_l \in (\mathcal{X}_g)^{\bullet} \bigcap {}^{\bullet}(\mathcal{X}_i)$ at the date $\mathcal{A}^{+}_{gi} \backslash \mathcal{X}_j(h_{max})$.

However, the firing of transition i is still possible if a new firing of each transition $g \in G$ produces another token. This can be expressed by a shift in the numbering of the events. Therefore, relation $\mathcal{A}^{-}_{ig} \otimes \mathcal{X}_g(k) \leq \mathcal{X}_i(k+1) \leq \mathcal{A}^{+}_{gi} \backslash \mathcal{X}_g(k)$ for $k < h_{max}$, becomes relation $\mathcal{A}^{-}_{ig} \otimes \mathcal{X}_g(k+1) \leq \mathcal{X}_i(k+1) \leq \mathcal{A}^{+}_{gi} \backslash \mathcal{X}_g(k+1)$ for $k \geq h_{max}$, in the new algebraic model.

Example 1 continued.

For $\mathcal{X}^{-}_0 = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}^t$, Theorem 3 provides the lowest trajectory $\mathcal{X}^{-}$ which is also trajectory $\mathcal{X}$ given in the first table of the example. Using these dates, we can deduce that the date of the first token death is 13. The new model is as follows : for $k \geq 2$, matrices $\mathcal{B}^{=}$, $\mathcal{B}^{-}$ and $\mathcal{B}^{+}$, replace the previous one in system (2.13).

$$\mathcal{B}^{=} = \begin{pmatrix} \varepsilon & \varepsilon & -2 \\ \varepsilon & \varepsilon & \varepsilon \\ 1 & \varepsilon & \varepsilon \end{pmatrix}, \mathcal{B}^{-} = \begin{pmatrix} \varepsilon & \varepsilon & 3 \\ 3 & \varepsilon & \varepsilon \\ \varepsilon & 6 & \varepsilon \end{pmatrix} \text{ and } \mathcal{B}^{+} = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ -11 & \varepsilon & \varepsilon \end{pmatrix}. \ \blacksquare$$

2.7.4 Computational complexity

The following curve gives indications on the possible CPU times needed to compute the different matrices w_k, and the lowest and greatest trajectories on an ordinary *Pentium* 1.3 GHz for a horizon h= 100. Computation tests are made using maxplus toolboxes under Scilab. The matrices $\mathcal{A}^{-}$ and $\mathcal{A}^{+}$ are completely full : there is a place containing a token between each couple of transitions. For instance, if $n = 50$, the relevant Petri net contains 50 transitions and 2500 places. The matrix $\mathcal{A}^{-}$ is generated randomly and $\mathcal{A}^{+}$ is deduced from $\mathcal{A}^{-}$ such that the system is temporally live on the desired horizon : the complete calculations are made, therefore. In that objective, we also take $\mathcal{A}^{=} = \varepsilon$ which do not effect significantly the time. At the moment, the code is not completely optimized and contains redundant operations.

The algorithms use elementary operations on matrices like $\otimes$, $\oplus$, $\backslash$, $\wedge$ and the more complex operation Klenne Star $*$. The last one determines the computational complexity of each step and the complexities of the different known algorithms are polynomial. Therefore, the complexity of calculation of the greatest trajectory is about $O(h.n^2)$ with h the horizon and n the dimension of the matrices. The space needed for the matrices w_k is $l.n^2$ with l the minimum of the horizon h and the length of the transient period. In short, the algorithm can consider important sizes of Event Graphs and horizon of calculation. Future papers will also consider sparse matrices.

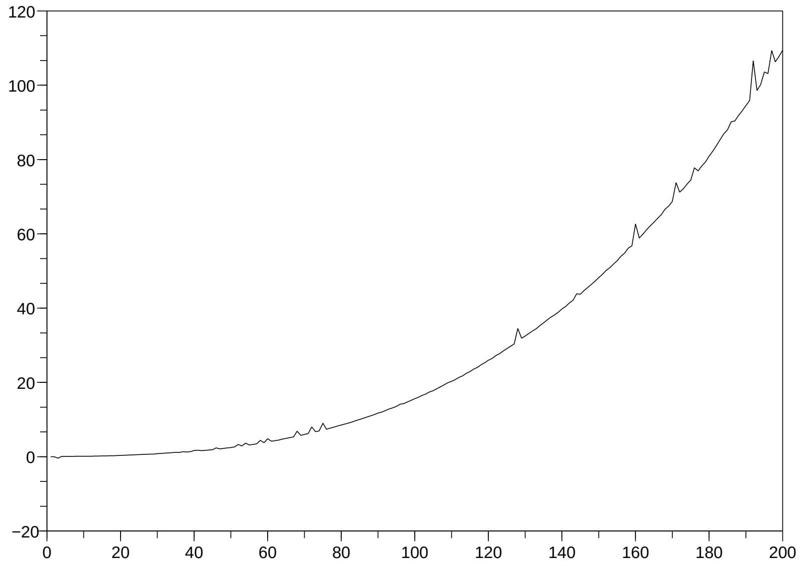

FIG. 2.8 : CPU time for different dimensions from 3 to 200 and h=100

2.8 Conclusion

Considering the (max, +) model of P-time Event Graphs, our first objective is the determination of the extremal state trajectories satisfying an initial condition defined on an interval. A direct resolution needs the storage of a matrix whose dimension is (h.nXh.n) and must follow the different forward/backward relations. Based on a specific series of matrices, the proposed resolution is composed of three steps : using the Kleene star, the iterative calculation determines the values of the greatest paths for different horizons ; a forward iteration generates a box containing the extremal trajectories ; a backward iteration gives the extremal trajectories. The introduction of a nondecreasing series of matrices alleviates the storage as the dimension is the size of the model, which depends on the number of the transitions and the initial marking. Therefore, each calculation processes reduced matrices of dimension $(n \times n)$. The approach can be applied to important processes for large horizons because the algorithms are strongly polynomial : the complexity is $O(h.n^3)$ for a given horizon h if the complexity of the used algorithm of Kleene star is $O(n^3)$ and we give the CPU time for different dimensions and horizons. Moreover, the results obtained in step a) for a given horizon can be reused in the calculations for a new horizon. The same remark can be made for step b) if the initial starting point is identical.

The determination of the maximal horizon of temporal consistency is the second objective. The technique is based on the analysis of convergence of matrices w_k^* : each entry can converge to a stable finite value or the infinite value $+\infty$. For a given P-time Event Graph, the case of a convergence to a constant matrix after a transitory period h_{max}, facilitates the storage and the reuse in the calculation of a new trajectory for any horizon. If the system is only consistent on horizon h_{max}, a non-synchronization cannot be avoided at $h_{max} + 1$ and we calculate the date of the first token deaths.

References

[1] Amari, S. 2005. Commande des graphes d'événements temporisés sous contraintes temporelles. *PhD Thesis*. University of Nantes, France, October 2005.

[2] Aygalinc, P., and Calvez, S., and Khansa, W., and Collart-Dutilleul, S. 1997. Using P-time Petri Nets for robust control of manufacturing systems. *1st IFAC-Workshop on Manufacturing Systems (MIM'97)*. Wien, Austria, pp.75-80.

[3] Baccelli, F., and Cohen, G., and Olsder, G.J., and Quadrat, J.P. 1992. *Synchronization and linearity. An Algebra for Discrete Event Systems*, available from http ://maxplus.org, New York, Wiley.

[4] B. Berthomieu and M. Diaz, Modeling and verification of time dependent systems using time Petri nets, IEEE Transactions on Software Engineering, 17(3), 1991.

[5] B. Berthomieu, D. Lime, O.H. Roux and F. Vernadat, Reachability Problems and Abstract State Spaces for Time Petri Nets with Stopwatches, Journal of Discrete Event Dynamic Systems, No. 17, pp. 133-158, 2007

[6] Cochet-Terrasson, J., and Gaubert, S., and Gunawardena, J. 1999. A constructive fixed point theorem for min-max functions. *Dynamics and Stability of Systems*. Vol. 14 N°4 : 407-433.

[7] Collart-Dutilleul, S., and Yim, P. 2003. Time window specification and validation with Petri nets. *ETFA , IEEE Conference on Emerging Technologies and Factory Automation*, Lisbon, Portugal, pp.232-237.

[8] Courtiat, J.P., and Diaz, M., and De Oliviera, R.C., and Sénac, P. 1996. Formal models for the description of timed behaviors of multimedia and hyper media distributed systems. *Computer Communications* 19 : 1134-1150.

[9] Declerck, P., and Didi Alaoui, M.K. 2005. Optimal Control synthesis in Interval Descriptor Systems, Application to Time Stream Event Graphs. *Proc. IFAC Congress* Praha, available from http ://www.istia.univ-angers.fr/%7Edeclerck/publications.html.

[10] Declerck, P., and Didi Alaoui, M.K. 2005. Extremal trajectories in P-time Event Graphs : application to control synthesis with specifications. *Proc. 44th IEEE Conference on Decision and Control and European Control Conference ECC, CDC-ECC'05*. Seville, Spain, pp.7621-7626, available from http ://www.istia.univ-angers.fr/%7Edeclerck/publications.html.

[11] Declerck, P., and Didi Alaoui, M.K. 2006. Liveness and acceptable trajectories in P-time Event Graphs, *Wodes 2006*, Ann Arbor Michigan, USA, 10-12 juillet 2006, available from http ://www.istia.univ-angers.fr/%7Edeclerck/publications.html.

[12] Declerck, P., Guezzi A. , Boimond J.-L. . Cycle Time of P-time Event Graphs. *4th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2007. Special Session : From Planning to Control of Manufacturing Systems.* Angers, France, May 09-12, 2007, available from http ://www.istia.univ-angers.fr/%7Edeclerck/publications.html.

[13] Declerck P. and M.K. Didi Alaoui, Modelling and liveness analysis of P-time event graphs in the (min, max, +) algebra, *IEEE International Conference on Systems, Man & Cybernetics*, The Hague, The Netherlands, October 10-13, 2004, available from http ://www.istia.univ-angers.fr/%7Edeclerck/publications.html.

[14] P. Declerck, A. Guezzi and C. Gros. *Temps de cycle des Graphes d'Evénements Temporisés et P-temporels.* Conférence Internationale Francophone d'Automatique, CIFA'2008, Bucharest, Romania, 2008.

[15] P. Declerck and M.K. Didi Alaoui, Optimal control synthesis of Timed Event Graphs with specifications defined by an interval model, *IEEE Transactions on Automatic Control*, February 2010.

[16] F. Defossez, S. Collart-Dutilleul and P. Bon, *Temporal requirements checking in a safety analysis of railway systems*, FORMS/FORMAT 2008, Symposium on Formal Methods for Automation and Safety in Railway and Automotive Systems, TU Braunschweig and Budapest University of Technology and Economics, October, 2008.

[17] Denat, J.P., Collart-Dutilleul, S. and Chetouane, F. 2005. Cycle time calculation for electroplating lines steady state operation using P-time Strongly Connected Event Graphs. *Proceedings of IMACS.* Paris, France, 2005.

[18] Diaz, M., and Owezarski, P. 1997. From multimedia models to multimedia transport protocols. *Computer Networks and ISDN Systems* 29 : 745-758.

[19] Didi Alaoui, M.K. 2005. Etude et supervision des graphes d'événements temporisés et temporels : vivacité, estimation et commande. *PhD Thesis.* University of Angers, France, available from http ://www.istia.univ-angers.fr/%7Edeclerck/resume-these-khalid.htm

[20] Gaubert, S. and Gunawardena, J. 1998. The duality theorem for min-max functions. *CRAS* t. 326, Série I, pp43-48.

[21] Khansa W., Denat J-P. and Collart Dutilleul S., P-time Petri nets for manufacturing systems. *WODES'96*, Edinburgh, UK, 1996.

[22] Kumar, R., and Garg, V.K. 1995. Extremal solutions of inequations over lattices with applications to supervisory control. *Theoret. Comp. Sci.* 148 : 67-92.

[23] Libeaut, L. 1996. Sur l'utilisation des dioïdes pour la commande des systèmes à événements discrets. *Phd Thesis.* Ecole Centrale de Nantes, France.

[24] M. Magnin. Réseaux de Petri à chronomètres : temps dense et temps discret, *PhD Thesis*, Ecole Centrale de Nantes, France, December 2007.

[25] Mc Millan K. and Dill D. Algorithms for interface timing verification, *Proceedings of the IEEE, International Conference on Computer Design : VLSI in Computers and Processors*, 1992.

[26] Walkup E., Optimization of linear max-plus systems with application to timing analysis, University of Washington, *PhD Thesis*, 1995.

[27] Van der Woude, J. and Olsder, G. J. 2005. On (min,max,+)-inequalities. *Contemporary Mathematics*, Vol. 377 : 353-361.

Chapitre 3

Temps de cycle

Dans cette partie, nous considérons un nouveau modèle dont nous analysons le temps de cycle et effectuons une optimisation liée aux durées des tâches. Ce nouveau modèle est un graphe d'événements P-temporel présentant des dépendances entre tâches. L'utilisation de la théorie des chemins classiques (Ford, Bellman,...) et l'algèbre (max, +) avec le théorème de Karp, ne pouvant être effectuée directement, nous exploitons la théorie de la programmation linéaire. Les résultats aboutissent également à une analyse de la vivacité.

3.1 Introduction

Le fonctionnement régulier des entreprises, des réseaux de transport sont autant d'exemples où l'existence d'horaires réguliers conditionne la bonne marche du système. Autour de cette caractéristique, différentes techniques se sont développées afin d'augmenter le taux de production, d'optimiser les ressources nécessaires, ... Critère temporel, le temps de cycle équivalent au taux de production, permet de caractériser la vitesse des systèmes et par exemple la production des systèmes industriels.

Pour les graphes d'événements, une approche est de provoquer le régime périodique en choisissant de manière judicieuse la première date de tir de chaque transition. On obtient ainsi un fonctionnement périodique de la forme suivante $x_i(k+1) = x_i(k) + \lambda$ avec $x_i(k)$ la k-ième date de tir de la transition x_i. Ainsi, le tir de chaque transition arrivera régulièrement selon la période λ.

Rappelons que pour un graphe d'événements temporisé fonctionnant de manière libre mais à vitesse maximale, le comportement périodique est tout à fait naturel. Cependant, il n'apparaît qu'après un transitoire qui peut être extrêmement long et qui correspond à un démarrage du système [1]. La longueur de ce transitoire n'est pas uniquement dépendante de la dimension du système car des systèmes de dimension réduite peuvent présenter un transitoire important. Ainsi, pour une durée de fonctionnement donnée, par exemple la durée d'une journée d'une entreprise, le transitoire pourra limiter la représentativité du classique taux de production car celui-ci n'est défini que par rapport à un comportement périodique.

Dans ce chapitre, nous allons montrer que le concept de périodicité et de temps de cycle existe également pour un modèle généralisé qui est celui d'un Graphe d'Evénement P-temporel mais présentant des interdépendances entre des durées de séjour de jetons. Un problème est celui de la maximisation/minimisation du temps de cycle mais aussi des durées de séjour de jetons. Nous donnons des conditions d'existence et des techniques de calcul résolvant ces problèmes. Nous montrons que le problème de la maximisation/minimisation du temps de cycle ne peut pas se réduire à l'analyse classique des circuits d'un graphe associé mais nécessite l'emploi d'autres outils comme la programmation linéaire.

3.2 Objectives

In this chapter, we propose a generalization of this model by introducing links between residence durations. Indeed, in some practical examples, some tasks must compensate for the undesirable effects of other operations like the warming of a part or an incomplete achievement. The treatment of this new Petri net needs its modeling and we will describe its algebraic model in the standard algebra. We will show that the relevant matrices generalize the incidence matrices obtained in previous works [5] [6].

The second objective is the performance analysis. Classically, the cycle time (or equivalently, the production rate) is based on the consideration of the circuits and determined by the calculation of the maximum of the ratios defined by the sum of temporisations to the sum of the number of the initial tokens, for each elementary circuit [15]. However, this technique cannot be applied on our new model as the considered matrices are not incidence matrices and cannot be associated with a graph. The well-known Theorem [15] based on the elementary circuits of the associated graph and the algorithm of Karp, cannot be applied. Therefore, we propose to use linear programming to solve this new problem which includes the initial one. We assume that the trajectory follows a 1-periodic behavior which facilitates the detection of the perturbations by the supervisory control.

A third objective is the optimisation of the cycle time but also of the residence duration of some tasks. Indeed, we focus on a periodic trajectory leading to an optimal behavior of the tasks where some residence duration are minimized and/or maximized. Practical examples are the minimization of the duration of a task which leads to energy savings, the minimization of the working time of the employees which has to be paid, or the maximization of the use or the availability of a machine.

At our knowledge, the papers considering cycle time or production rate with linear programming [12] [2] [3] [8] [13] do not use the theorems of Farkas and Stiemke. We also consider in this chapter the optimization of a general criterion which includes the cycle time but is not limited to. Moreover, the considered Petri net is original as it is more general than the Timed Event Graph and the P-time Event Graph. Similar but different to the algebraic model [11] based on numbers of transition firings ("counter" approach), our algebraic model focus on the dates of transition firings and a first version has been introduced in [5]. Note that this algebraic model can also be used in control [9].

The chapter is organized as follows. Section 3.3 defines the generalized P-time Event Graphs and its algebraic model. Using a variant of Farkas' lemma, Section 3.4.4 gives conditions of consistency of the cycle time. Then we propose two techniques of linear programming allowing the determination of the cycle times and the dates of transitions firing. A simple example illustrates the different concepts. The reader can find in Appendix the technique allowing the reduction of the horizon associated with the model.

3.3 Preliminary

A Petri net is a pair (G, M_0), where $G = (R, V)$ is a bipartite graph with a finite number of nodes (the set V) which are partitioned into the disjoint sets of places P and transitions TR (transitions are denoted x while temporization are denoted T, T^- or T^+); R consists of pairs of the form (p_i,x_i) and (x_i,p_i) with $p_i \in P$ and $x_i \in TR$. The initial marking M_0 is a vector of dimension $\mid P \mid$ whose elements denote the number of initial tokens in the respective places. The set ${}^\bullet p$ is the set of input transitions of p and $p^\bullet$ is the set of output transitions of place $p \in P$. The set ${}^\bullet x_i$ (respectively, $x_i^\bullet$) is the set of the input (respectively, output) places of the transition $x_i \in TR$.

For a Petri net with $\mid P \mid$ places and $\mid TR \mid$ transitions, the incidence matrix $W = [W_{ij}]$ is a $\mid P \mid \times \mid TR \mid$ matrix of integers and its entry is given by $W_{ij} = W_{ij}^+ - W_{ij}^-$ where W_{ij}^+ is the weight of the arc from the transition j to the place i and W_{ij}^- is the weight of the arc from the place i to the transition j [14].

In a Petri net, a firing sequence from a marking M, implies a string of successive markings. The characteristic vector s of a firing sequence S is such that each component is an integer corresponding to the number of firings of the corresponding transition. A marking M reached from initial marking M_0 by firing of a sequence S can be calculated by the fundamental relation : $M = M_0 + W \times s$.

Definition 4. *A Petri net is called an Event Graph if each place has exactly one upstream and one downstream transition.*

P-time Petri nets allow the modeling of discrete event systems with sojourn time constraints of the tokens inside the places. Consistent with the dioid $\overline{\mathbb{R}}_{max}$ (see [1]), we associate a temporal interval defined in $\mathbb{R}^+ \times (\mathbb{R} \cup \{+\infty\})$ with each place. Each place $p_l \in P$ is associated with an interval $[T_l^-, T_l^+]$, where T_l^- is the lower bound and T_l^+ the upper bound. Its initial marking is denoted $(M_0)_l$.

Definition 5. *A P-time Event Graph is a triplet* (G, M_0, g) *where* G *is an Event Graph,* M_0 *is the initial marking and the mapping* g *is defined by* $p_l \rightarrow [T_l^-, T_l^+]$ *with* $0 \leq T_l^- \leq T_l^+$ *from* P *to* $\mathbb{R}^+ \times (\mathbb{R}^+ \cup \{+\infty\})$.

The interval $[T_l^-, T_l^+]$ is the static interval of duration time of a token in place p_l. The token must stay in this place during the minimum residence duration T_l^-. Before this duration, the token is in a state of unavailability to fire the outgoing transition. The value T_l^+ is a maximum residence duration after which the token must leave the place p_l (and can contribute to the enabling of the downstream transition). If not, the system falls into a token-dead state. So, the token is available to fire the outgoing transition in the time interval $[T_l^-, T_l^+]$.

3.4 Problem wihout optimization

3.4.1 Objective

We consider the "dater" type well-known in the (max, +) algebra : each variable $x_i(k)$ represents the date of the k^{th} firing of transition x_i. Let m be the maximum number of initial tokens : $m = max\{(M_0)_l \mid l \in [1, \mid P \mid]\}$. The objective is now the analysis of a 1-periodic trajectory defined by $x(k+1) = \lambda \times u + x(k)$ where λ is the cycle time and $u = (1, 1, ..., 1)^t$, using the following algebraic model

$$\begin{pmatrix} G^- \\ G^+ \end{pmatrix} \times \begin{pmatrix} x(k-m) \\ x(k-m+1 \\ \vdots \\ x(k-1) \\ x(k) \end{pmatrix} \leq \begin{pmatrix} -T^- \\ T^+ \end{pmatrix} \tag{3.1}$$

where the dimension of $G^- = [G_m^- G_{m-1}^- G_{m-2}^- \cdots G_1^- G_0^-]$ and $G^+ = [G_m^+ G_{m-1}^+ G_{m-2}^+ \cdots G_1^+ G_0^+]$ is equal to $\mid P \mid . \ (m+1). \mid TR \mid$.

Without reduction of generality, we suppose in the sequel that $m = 1$ (see Appendix). Below, we show that a P-temporal event graph with interdependent residence durations can be expressed with the previous form. The existence of a solution and the optimization of a specific criterion will be described in the sequel.

3.4.2 Matrix expression of a P-time Event Graph

Let us express the firing interval for each transition of P-time Event Graphs, guaranteing the absence of token-dead states. If we assume a FIFO functioning of the places which guarantees that the tokens do not overtake one another, a correct numbering of the events can be carried out. The evolution can be described by the following inequalities expressing relations between the firing dates of transitions. Let us recall that an Event Graph can be considered as a set of connected subgraphs made up of a place p_l linked with one upstream transition $\{x_j\} =^\bullet p$ and one downstream transition $\{x_i\} = p^\bullet$.

Using the lower bound T_l^- and the upper bound T_l^+, we can write the following system for each place p_l where $(j, i) = ({}^\bullet p, p^\bullet)$:

$$\begin{cases} x_j(k - (M_0)_l) - x_i(k) \leq -T_l^- \\ -x_j(k - (M_0)_l) + x_i(k) \leq T_l^+ \end{cases} .$$

- In the first row (respectively, second row), the weight $+1$ (respectively, -1) of $x_j(k-(M_0)_l)$ is the weight of the arc going from x_j to place p_l which is equal to $+W_{lj}^+$ (respectively, $-W_{lj}^+$).
- In the first row (respectively, second row), the weight -1 (respectively, $+1$) of $x_i(k)$ is the weight of the arc going from place p_l to transition x_i which is equal to $-W_{li}^-$ (respectively, $+W_{li}^-$).

The set of the previous inequalities which describes a P-time Event Graph, can be expressed with the previous symmetrical form (3.1) : Column-vectors $-T^-$ and T^+ are vectors of temporization where $[T_l^-, T_l^+]$ is the time interval of place p_l. Each place corresponds to a row of G^- which contains the weights of its entering and outgoing arcs which are usually expressed by W^+ and W^-.

- The entry (l, j) of the matrix G_r^- for $r \in [0, m]$ contains the weight with the sign plus, of the arc going from the transition x_j to the place p_l with an initial marking $(M_0)_l = r$.
- Moreover, the entry (l, j) of the matrix G_0^- contains the weight with the sign minus, of the arc from the place p_l to the transition x_j , for any initial marking $(M_0)_l \in [0, m]$.

So, $G_r^- \geq 0$ for $r \in [1, m]$ and a coefficient of the matrix G_0^- can be null, negative or positive : $G_0^- \geq -W^-$. The interpretation of the matrix G^+ is identical to G^- but with a change of sign of the coefficients : $G^+ = -G^-$.

The notation $M_{i,.}$ corresponds to the row i of the matrix M. From the above description on the weight of the arcs, we can deduce the following relation expressing the incidence matrix W :

$$W_{l,.} = \sum_{r=0}^{m} (G_r^-)_{l,.} = -\sum_{r=0}^{m} (G_r^+)_{l,.} \tag{3.2}$$

for each place l of the P-temporal event graph.

- Note that $G_1^- = W^+$ and $G_0^- = -W^-$ if the initial marking of each place is unitary.
- Note that $G_0^- = W$ and there is no G_1^-, if the initial marking is null ($M_0 = 0$).

3.4.3 Matrix expression of a P-time Event Graphs with interdependent residence durations.

We can also consider the case of residence duration of a token in place p_l which determines the temporisation of another place. For instance, a product which has been cooked cannot be put immediately in a package and needs to cool down. In food industry, specific cooling systems are used. We can suppose the temporisation of the cooling of a part is an affine relation depending of its cooking time. These residence durations are called interdependent.

We assume that the initial marking is null : a realistic assumption is that not part stays in the oven and the colling system initially. Naturally, the following relations can be rewritten with a non-null initial marking. Using lower bound T_l^- we can write the following inequality for a place p_l where $(x_j, x_i) = (^\bullet p_l, p_l^\bullet)$:

$T_l^- + x_j(k) \leq x_i(k)$.

Let us consider another place p_h where $(x_{j'}, x_{i'}) = (^\bullet p_h, p_h^\bullet)$: the residence duration of a token is $x_{i'}(k) - x_{j'}(k)$.

Now, we say that the place p_l is dependant from the place p_h with an affine function if we can write $T_l^- = a.(x_{i'}(k) - x_{j'}(k)) + b$. Note that b is coherent with a temporization.

Therefore, the obtained inequation for place p_l is $a.(x_{i'}(k) - x_{j'}(k)) + x_j(k) - x_i(k) \leq -b$ which is relation between four variables. This relation corresponds to a row in G_0^- with the entries a, $-a$, 1 and -1 ; the relevant entry in $-T^-$ is $-b$; symetrically, the entries in G_0^+ and T^+ are null. When the outstreaming transition $x_{i'}$ of the place p_h is the instreaming transition x_j of the place p_l, the relation becomes which is a relation between three variables. We have $(1 + a).x_{i'}(k) - a.x_{j'}(k) - x_i(k) \leq -b$.

Containing more variables, these relations are more complex that the relations of the P-time Event Graphs in their classic form and the coefficients depends on the entries of the incidence matrix but also of the coefficients of the affine function.

3.4.4 General form $Ax \leq b$

We now start the resolution of the problem and deduce from (3.1) the following system

$$\begin{pmatrix} G_1^- & G_0^- \\ G_1^+ & G_0^+ \end{pmatrix} . \begin{pmatrix} x(k) \\ \lambda.u + x(k) \end{pmatrix} \leq \begin{pmatrix} -T^- \\ T^+ \end{pmatrix}$$

where

$$\begin{pmatrix} (G_1^- + G_0^-) \\ (G_1^+ + G_0^+) \end{pmatrix} .x + \begin{pmatrix} G_0^- \\ G_0^+ \end{pmatrix} .\lambda.u \leq \begin{pmatrix} -T^- \\ T^+ \end{pmatrix} \tag{3.3}$$

If we simplify the writing with $x(k) = x$, we obtain the following system which presents the general form $Ax \leq b$,

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} . \begin{pmatrix} x \\ \lambda \end{pmatrix} \leq \begin{pmatrix} -T^- \\ T^+ \end{pmatrix} \tag{3.4}$$

where $A_{11} = G_1^- + G_0^-$, $A_{12} = G_0^-.u$, $A_{21} = G_1^+ + G_0^+$ and $A_{22} = G_0^+.u$.

Let us consider the important particular case of a P-temporal event graph. The following property makes the connection with the incidence matrices.

Property 3. *For a P-time Event Graph, we have* $A_{11} = W = -A_{21}$, $A_{12} = -M_0 = -A_{22}$ *and system (3.4) becomes*

$$\begin{pmatrix} W & -M_0 \\ -W & M_0 \end{pmatrix} \cdot \begin{pmatrix} x \\ \lambda \end{pmatrix} \leq \begin{pmatrix} -T^- \\ T^+ \end{pmatrix}. \tag{3.5}$$

Proof.

This writing can be rewritten because we deduce from (3.2) $G_1^- + G_0^- = W$ and $G_1^+ + G_0^+ = -W$ .So, we can deduce that $A_{11} = W$, $A_{21} = -W$

Let us note that $G_0^- = -W^-$ and $G_1^- = W^+$ when the P-time Event Graph has one token by place initially. Thus, $G_0^-.u = -W^-.u = -M_0$ which represents the initial marking. This result is also true when $m = 1$, that is, each place initially has one token at the most. Indeed, each place without token is represented by a line of G_0^- which contains the weights 1 and -1 corresponding to its ingoing and outgoing arcs. The similar reasoning holds for the lower part of the system and we obtain $G_0^+.u = W^-.u = M_0$. ■

3.4.5 Example

Model

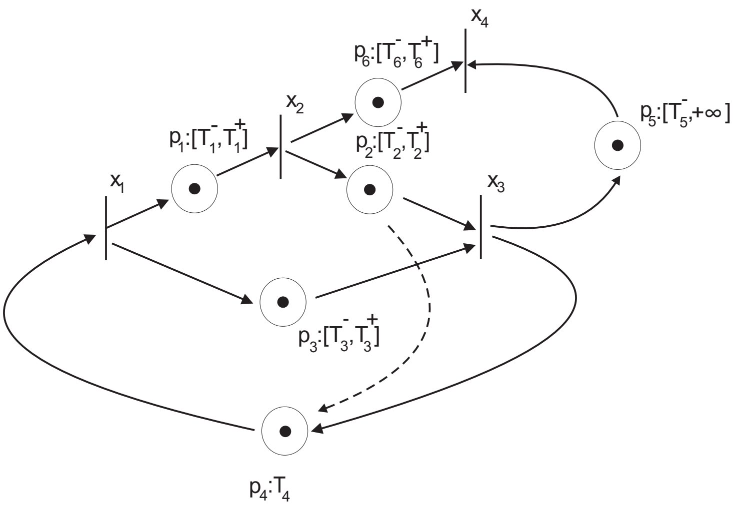

Fig. 3.1 : P-time Event graph with interdependent residence durations (dotted line)

The state is $x(k) = \begin{pmatrix} x_1(k) & x_2(k) & x_3(k) & x_4(k) \end{pmatrix}^t$. The temporal intervals are : $[T_1^-, T_1^+] = [3, 10]$, $[T_2^-, T_2^+] = [3, 20]$, $[T_3^-, T_3^+] = [1, 2]$, $[T_5^-, +\infty] = [11.5, +\infty]$ and $[T_6^-, T_6^+] = [1, 5]$. The place p_4 describes the cold down of the product which has been cooked in an oven (place p_2) : $T_4 = a.(x_3(k) - x_2(k-1)) + b$ with $a = 5$ and $b = 3$. The matrices of the algebraic model are as follows.

$T^- = \begin{pmatrix} 3 & 3 & 1 & b & 11.5 & 1 \end{pmatrix}^t$ and $T^+ = \begin{pmatrix} 10 & 20 & 2 & +\infty & +\infty & 5 \end{pmatrix}^t$

$$G_1^- = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & -a & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}, G_0^- = \begin{pmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 \\ -1 & 0 & a & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 \end{pmatrix},$$

$$G_1^+ = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix} \text{ and } G_1^- = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

We now deduce that $A_{11} = \begin{pmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 1 & 0 & -1 & 0 \\ -1 & -5 & 6 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 1 & 0 & -1 \end{pmatrix}$, $A_{12} = \begin{pmatrix} -1 \\ -1 \\ -1 \\ 4 \\ -1 \\ -1 \end{pmatrix}$, $A_{21} = \begin{pmatrix} -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 \end{pmatrix}$

and $A_{22} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$. ■

3.4.6 Existence of a 1-periodic behavior

After the obtention of a linear inequalities system having the form $A.x \leq b$,, we prove the existence of a 1-periodic behavior using a known theorem in linear programming.

As the vector b is finite, we assume that each infinite bound $T_l^+ = +\infty$ is replaced by an arbitrarily large size finite what neutralizes the constraint $x_i(k) - x_j(k - m_l) \leq T_l^+ = +\infty$.

Corollary 3. *Farkas' lemma (variant) Corollary 7.1.e in [16] Let A be a matrix and let b a vector. Then the system $A \times x \leq b$ of linear inequalities has a solution x, if and only if $y \times b \geq 0$ for each row vector $y \geq 0$ with $y \times A = 0$*

Theorem 7. *The system (3.1) with $m = 1$ can follow a 1-periodic behavior for a given cycle time λ, if and only if, for each row vector $y \geq 0$ with*

$$y \times \begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} = 0, \tag{3.6}$$

–

$$y \times \begin{pmatrix} -T^- \\ T^+ \end{pmatrix} \geq 0 \text{ if } y \times \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix} = 0. \tag{3.7}$$

– *and the following lower and upper bounds are consistent : y is associated with the lower bound of* λ

$$\frac{y \times \begin{pmatrix} -T^- \\ T^+ \end{pmatrix}}{y \times \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix}} \text{ if } y \times \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix} < 0, \tag{3.8}$$

y is associated with the upper bound of λ

$$\frac{y \times \begin{pmatrix} -T^- \\ T^+ \end{pmatrix}}{y \times \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix}} \text{ if } y \times \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix} > 0, \tag{3.9}$$

Proof.

From Farkas' lemma, we can deduce that the system $\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \cdot \begin{pmatrix} x \\ \lambda \end{pmatrix} \leq \begin{pmatrix} -T^- \\ T^+ \end{pmatrix}$ $\begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} .x \leq \begin{pmatrix} -T^- \\ T^+ \end{pmatrix} - \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix} .\lambda$ has a solution x, if and only if $y \times (\begin{pmatrix} -T^- \\ T^+ \end{pmatrix} - \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix} .\lambda) \geq 0$ for each row vector $y \geq 0$ with $y \times \begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} = 0$.

So, $y \times \begin{pmatrix} -T^- \\ T^+ \end{pmatrix} \geq y \times \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix} .\lambda = \lambda \times y \times \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix}$. The determination of λ leads to the upper bound $\frac{y \times \begin{pmatrix} -T^- \\ T^+ \end{pmatrix}}{y \times \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix}} \geq \lambda$ if $y \times \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix} > 0$, the lower bound $\frac{y \times \begin{pmatrix} -T^- \\ T^+ \end{pmatrix}}{y \times \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix}} \leq \lambda$ if $y \times \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix} < 0$, and $y \times \begin{pmatrix} -T^- \\ T^+ \end{pmatrix} \geq 0$ if $y \times \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix} = 0$.

For any vectors y_1, y_2 satisfying $y \times \begin{pmatrix} W \\ -W \end{pmatrix} = 0$ and corresponding to the lower and upper bounds respectively, we have $\frac{y_1 \times \begin{pmatrix} -T^- \\ T^+ \end{pmatrix}}{y_1 \times \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix}} \leq \frac{y_2 \times \begin{pmatrix} -T^- \\ T^+ \end{pmatrix}}{y_2 \times \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix}}$ as λ is defined by these bounds. ■

Therefore, the theorem brings a way of checking the consistency of (3.4) and the existence of a 1-periodic trajectory for the extended P-time Event Graphs. Moreover, this result gives a frame of the cycle time given by the maximum of the lower bounds and the minimum of the upper bounds. Note that the existence of a solution depends on λ in the two first relations contrary to the last one.

Remark 8. *Note that the vector y is not a P-invariant as its dimension is $2. \mid P \mid$. It is a generalized P-invariant which depends on the incidence matrix expressed in A_{11} and A_{21} but also on the affine functions of the interdependent residence durations.*

Graphical interpretation. In the case of a P-time Event Graph, equation (3.6) becomes $y \times \begin{pmatrix} W \\ -W \end{pmatrix} = 0$ and a graph can be associated with matrix $\begin{pmatrix} W \\ -W \end{pmatrix}$: As W is the incidence matrix of an event graph where each row contains the two entries 1 and -1 (corresponding to columns i and j) at the most, each row l of this matrix can be associated with an arc coming from vertex j to vertex i . Following the relation (3.5) , we can associate the time value $-T_l^-$ (and the marking value $-(M_0)_l = (A_{12})_l$) with the arc l coming from vertex i to vertex j for the upper matrix W and the time value T_l^+ (and the marking value $(M_0)_l = (A_{22})_l$) with the relevant arc coming from vertex j to vertex i for the lower matrix $-W$. As the non-null entries of the vector $y \geq 0$ can be normalized to 1 , the vector y highlights the rows and their bounds which correspond to the minimal and maximal solutions. It is well known that each vector y defines a circuit and that the set of these circuits describes the minimal and maximal critical structures which determines the bounds of the cycle time.

However, this graphical interpretation does not hold for a P-time Event Graph with interdependent residence durations : we have $A_{11} \neq W$ and $A_{21} \neq -W$ and a row cannot be associated with an arc as it can contain more than two entries. We can define a different associated graph expressing the connections between the rows and the columns of matrix $\begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix}$ as follows : each row is associated with a specific element (a black dot, for instance) linked to the different vertexes corresponding to the columns (a vertical line like a transition). Moreover, the entries of the vector y cannot be normalized to 1 in general, which implies that the selection of the rows by the non-null entries of vector y , and the relevant selected subgraph, presents a ponderation. If we neglect this ponderation, the selected rows do not define a circuit but a more complex structure. In fact, each vector y expresses a dependance between the rows of the relevant submatrix. Following the relation (3.4), we can associate the pair $(-T_l^-\ ,\ (A_{12})_l\ )$ with the row l of the upper matrix A_{11} and the pair $(T_l^+\ ,\ (A_{22})_l)$ with the row l of the lower matrix A_{21}. The following example shows that only a subset of the vectors y corresponds to a subset of circuits, in the Petri net or in the associated graph.

3.4.7 Example continued

More general than the matrices obtained in the studies [5] [6], the matrices G_1^- and G_0^- are almost but are not the incidence matrices W^+ and W^- of an event graph as the fourth row contains two non-null entries (see 3.4.5).

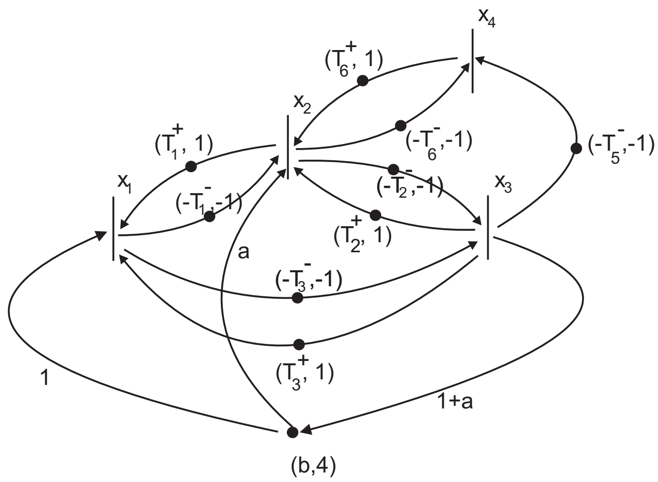

FIG. 3.2 : Associated graph : each row corresponds to a black dot while each column is expressed by a vertical line.

We also have $A_{11} \neq W$ and $A_{21} \neq -W$. The analysis of the independent row-vectors y

$$\begin{pmatrix} 0.316 & 2.025 & 0.025 & 0.341 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 2.5 & 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0.33 & 0 & 0 & 1.66 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

highlights the following relevant structures

y	structure	bound
y_1	$-T_1^-, -T_2^-, -T_3^-$ and b	lower bound : 8.075
y_2	$-T_2^-, -T_5^-$ and T_6^+	lower bound : 9.5
y_3	$-T_2^-, -T_3^-$ and b	lower bound : 9.5
y_4	$-T_3^-, b$ and T_1^+	upper bound : 13.66

.

The structure selected by y_2 ($-T_2^-$,$-T_5^-$ and T_6^+) corresponds to a circuit in the associated graph contrary to the structures selected by y_1, y_3 and y_4 which present non-disjoint circuits. Note that the structure selected by y_1 is not irreducible as the relevant sumatrix has four rows and three columns. Therefore, the usual theorem which gives the cycle time based on the circuits, does not hold. We obtain $\max(8.075, 9.5, 9.5) \leq \lambda \leq \min(13.66)$ and the interval of the possible values is $[9.5, 13.66]$.

Note also that the vectors y_2 and y_4 which define a lower and an upper bound respectively of the cycle time, select a combinaison of lower and upper bounds of temporisations and not only a set of lower (respectively, upper) bounds of temporisations. ■

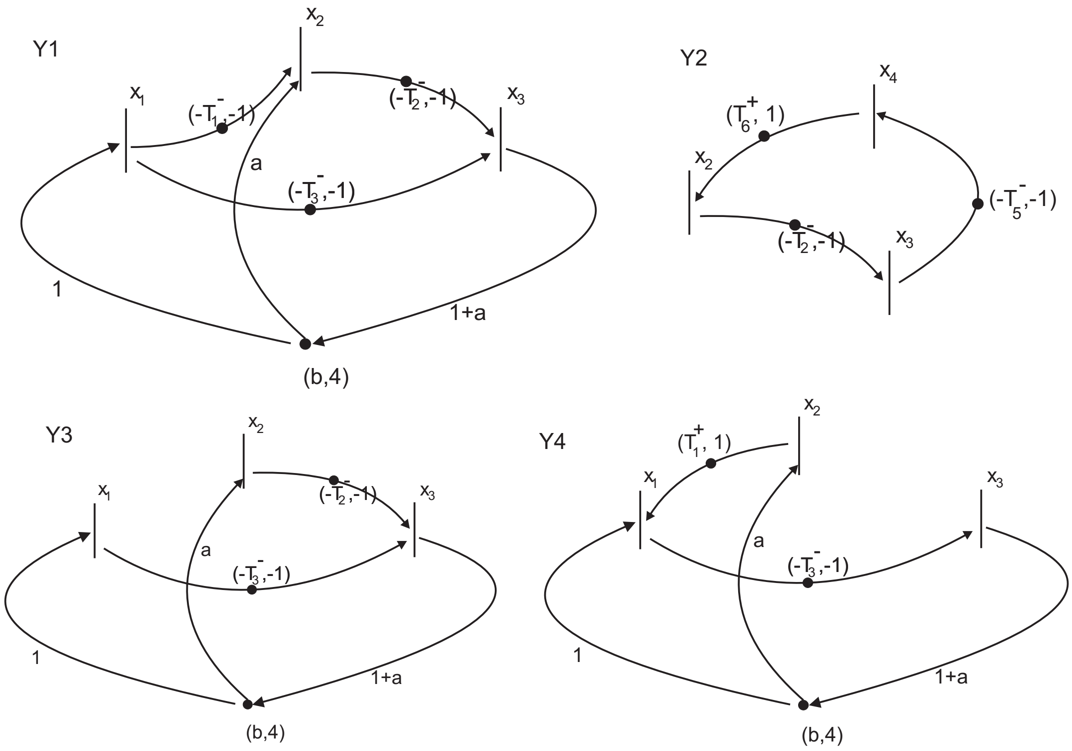

FIG. 3.3 : Substructures

3.5 Optimization

3.5.1 Approach 1

Let us define the criterion of our problem. A classical objective can be the determination of the minimal and maximal cycle time like in many studies. The objective can also be the maximisation of the residence duration of some tasks like the use of machines. It can be the minimisation of the residence duration of some other tasks like the working time of the employees. The sets of the places are denoted $P_{machines}$ and $P_{employees}$ respectively. Therefore, a more general criterion is $Cr.\begin{pmatrix} x \\ \lambda \end{pmatrix} = \alpha. \sum_{p_l \in P_{machines}} (x_{p_l^\bullet}(k) - x_{^\bullet p_l}(k - m_l)) + \beta. \sum_{p_l \in P_{employees}} (x_{p_l^\bullet}(k) - x_{^\bullet p_l}(k - m_l)) + \gamma.\lambda$ where α, β and γ are ponderation coefficients. We can focus on the optimisation of the residence durations : the goal is the maximisation of the use of the machines and the minimisaton of the working time of the employees during the pattern λ ; we take $\alpha > 0$, $\beta < 0$ and $\gamma = 0$. The periodic behavior leads to the repetion of the obtained gain.

As $x_{p_l^\bullet}(k) - x_{^\bullet p_l}(k - m_l) = x_{p_l^\bullet}(k) - x_{^\bullet p_l}(k) + m_l.\lambda$, a more simple writing is as follows :

$Cr.\begin{pmatrix} x \\ \lambda \end{pmatrix} = \alpha. \sum_{p_l \in P_{machines}} (x_{p_l^\bullet}(k) - x_{^\bullet p_l}(k)) + \beta. \sum_{p_l \in P_{employees}} (x_{p_l^\bullet}(k) - x_{^\bullet p_l}(k)) + \gamma'.\lambda$ where $\gamma' = \gamma + \alpha. \sum_{p_l \in P_{machines}} m_l + \beta. \sum_{p_l \in P_{employees}} m_l$.

A finite optimal solution yields a realistic trajectory and the smallest or largest vector x are not bounded up to now. The solution is to limit the minimization (respectively, maximization) by a minorant (respectively, a majorant) taken finished. The minorant of state x (respectively, majorant) is noted L (respectively, noted U). The non-decrease of the trajectories can be obtained by adding the relation $x(k+1) \geq x(k)$ in the model and a consequence is $\lambda \geq 0$. The case of an infinite greatest λ obtained in the linear program, shows that the cycle time is not upper bounded.

Under the condition of compatibility of the system of constraints, the optimal solution is given by the following linear programming problems.

$\min Cr.\begin{pmatrix} x \\ \lambda \end{pmatrix}$ under constraint :

$$\begin{pmatrix} -I & 0 \\ A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \cdot \begin{pmatrix} x \\ \lambda \end{pmatrix} \leq \begin{pmatrix} -L \\ -T^- \\ T^+ \end{pmatrix} \tag{3.10}$$

$\max Cr.\begin{pmatrix} x \\ \lambda \end{pmatrix}$ under constraint :

$$\begin{pmatrix} I & 0 \\ A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \cdot \begin{pmatrix} x \\ \lambda \end{pmatrix} \leq \begin{pmatrix} U \\ -T^- \\ T^+ \end{pmatrix} \tag{3.11}$$

Remark 9. *The rows corresponding to an arbitrary large upper temporisation T_i^+ can be removed as they do not modify the calculation.*

The reduction to the optimisation of the cycle time of a P-time Event Graph without interdependent residence durations, is as follows. The minimisation/maximisation of only the cycle time λ is obtained with $\alpha > 0$, $\beta = 0$ and $\gamma = 0$ and the constraints of a P-time Even Graph without interdependent residence durations, are given by $A_{11} = W = -A_{21}$ and $A_{12} = -M_0 = -A_{22}$. Note that each line of the constraint contains 3 variables per inequality(λ and 2 variables x_i) which is not the general case.

3.5.2 Example continued.

We below take $L = \begin{pmatrix} 0 & 0 & 0 & 0 \end{pmatrix}^t$ and $U = \begin{pmatrix} 20 & 20 & 20 & 20 \end{pmatrix}^t$.

Criterion 1 : We focus on the optimisation of the cycle time λ.

$\lambda_{\min}$: Criterion : $Cr = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 \end{pmatrix}$

We obtain $\begin{pmatrix} x_1 & x_2 & x_3 & x_4 & \lambda \end{pmatrix}^t = \begin{pmatrix} 8.5 & 6.5 & 0 & 2 & 9.5 \end{pmatrix}^t$ with the criterion value $f =$ 9.5. The minimal cycle time is $\lambda_{\min} = 9.5$. So, the relevant trajectory is $x(0) = \begin{pmatrix} 8.5 & 6.5 & 0 & 2 \end{pmatrix}^t \rightarrow$ $x(1) = \begin{pmatrix} 18 & 16 & 9.5 & 11.5 \end{pmatrix}^t \rightarrow x(2) = \begin{pmatrix} 27.5 & 25.5 & 19 & 21 \end{pmatrix}^t \rightarrow \ldots$

$\lambda_{\max}$:

We obtain $\begin{pmatrix} x_1 & x_2 & x_3 & x_4 & \lambda \end{pmatrix}^t = \begin{pmatrix} 20 & 16.33 & 7.33 & 7.66 & 13.66 \end{pmatrix}^t$ with $f = 13.66$. The maximal cycle time is $\lambda_{\max} = 13.66$.

Criterion 2 : Now we want : 1) to minimize the cycle time λ ; 2) to minimize $x_3(k) - x_2(k-1)$ which expresses the cooking of a product ; 3) to maximize $x_2(k) - x_1(k-1)$. So, we take $Cr. \begin{pmatrix} x \\ \lambda \end{pmatrix} = \alpha.(x_{p_2^\bullet}(k) - x_{^\bullet p_2}(k - m_2)) + \beta. \sum_{p_l \in P_{employees}} (x_{p_1^\bullet}(k) - x_{^\bullet p1}(k - m_1)) + \gamma.\lambda$ with the ponderations $\alpha = 2$, $\beta = -2$ and $\gamma = 1$. We obtain $Cr. = \begin{pmatrix} -\beta & \beta - \alpha & \alpha & 0 & \alpha + \beta + 1 \end{pmatrix} = \begin{pmatrix} 2 & -4 & 2 & 0 & 1 \end{pmatrix}$.

The minimisation of the above criterion gives $\begin{pmatrix} x_1 & x_2 & x_3 & x_4 & \lambda \end{pmatrix}^t = \begin{pmatrix} 9 & 8 & 0 & 0.5 & 11 \end{pmatrix}^t$ with $f = -3$. The cycle time $\lambda = 11$ is greater than the minimal value $\lambda_{\min} = 9.5$. The analysis of the terms $x_3(k) - x_2(k-1)$ and $x_2(k) - x_1(k-1)$ for the two criteria shows that the term $x_2(k) - x_1(k-1)$ has been increased (7.5 for criterion 1 and 10 for criterion 2) while the other one is identical.

The maximisation of the criterion 2 leads to the same results as the criterion 1 : $\begin{pmatrix} x_1 & x_2 & x_3 & x_4 & \lambda \end{pmatrix}^t = \begin{pmatrix} 20 & 16.33 & 7.33 & 7.66 & 13.66 \end{pmatrix}^t$ with $f = 3$. ■

3.5.3 Approach 2.

In the approach 1, the used variables are the cycle time and the first date of shooting of each transition. The principle of the duality allows the replacement of these variables by a new variable denoted y which gives information like in section 3.4.6, on the part of the system which is at the origin of the optimal cycle time. Moreover, this principle generates a new algorithm.

3.5.3.1 The two dual forms

Let us recall first the dual problems.

Problem (P) $\underset{y \in \mathbb{R}^n}{min}\ y.b$ under $y.A = c$ and $y \geq 0$

and

Problem (D) $\max_{x \in \mathbb{R}^m} c.x$ under $A.x \leq b$

Let us consider the two problems of the approach 1. It corresponds to the form (D) with the following correspondances for the minimization (3.10)

$$A = \begin{pmatrix} -I & 0 \\ A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, b = \begin{pmatrix} -L \\ -T^- \\ T^+ \end{pmatrix}, x = \begin{pmatrix} x \\ \lambda \end{pmatrix} \text{ and } c = -Cr.$$

For the maximization (3.11), we have

$$A = \begin{pmatrix} I & 0 \\ A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, b = \begin{pmatrix} U \\ -T^- \\ T^+ \end{pmatrix}, x = \begin{pmatrix} x \\ \lambda \end{pmatrix} \text{ and } c = +Cr.$$

We thus obtain the following dual problem which is also a linear problem of programming :

$$\min y.\begin{pmatrix} -L^t & -(T^-)^t & (T^+)^t \end{pmatrix}^t \text{ under } y.\begin{pmatrix} -I & 0 \\ A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} = -Cr \text{ with } y \geq 0 \text{ and}$$

$$\min y.\begin{pmatrix} U^t & -(T^-)^t & (T^+)^t \end{pmatrix}^t \text{ under } y.\begin{pmatrix} I & 0 \\ A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} = Cr \text{ with } y \geq 0.$$

The following theorem will be useful.The set $X^{ad} = \{x \in \mathbb{R}^{\mathrm{m}} | A.x \leq b\}$ is the set of the admissible solution to the dual problem. In the same way, $Y^{ad} = \{y \in (\mathbb{R}^+)^{\mathrm{n}} | y.A = c\}$

Theorem 8. *(Chapter 4 in [4]) :*

1. *if $y \in Y^{ad}$ et $x \in X^{ad}$ then $y.b \geq c.x$*
2. *if $\bar{y} \in Y^{ad}$, $\bar{x} \in X^{ad}$ and $y.b = c.x$ then $\bar{y}$ et $\bar{x}$ are the optimal solutions to (P) and (D), respectively.*

In the sequel, we focus on the optimisation of the cycle time in the case of an extended P-time Event Graph. The particular case of the P-time Event Graphs will be made in the remarks.

3.5.3.2 Optimisation of the cycle time

So, $Cr = \begin{pmatrix} 0 \ldots 0 & -1 \end{pmatrix}$. The application of the Theorem of duality will show that the optimal solution of the approach 2 (problem P) also gives the optimal cycle time. Let $b_1^t = \begin{pmatrix} -L^t & (-T^-)^t & (T^+)^t \end{pmatrix}$ and $b_2^t = \begin{pmatrix} U^t & (-T^-)^t & (T^+)^t \end{pmatrix}$ the two vectors b in problems (3.10) and (3.11). Let $\begin{pmatrix} x \\ \lambda \end{pmatrix} \in X^{ad}$ and $y \in Y^{ad}$, $\begin{pmatrix} x_{min} \\ \lambda_{min} \end{pmatrix}$ and y_{min} ($\begin{pmatrix} x_{max} \\ \lambda_{max} \end{pmatrix}$ and y_{max}, respectively) be the optimal solutions to the problem of minimisation (maximisation, respectively).

So we have the following result when the optimisation of only the cycle time is considered.

Theorem 9. *Let $Cr = \begin{pmatrix} 0 \ldots 0 & 1 \end{pmatrix}$.*

$-y.b_1 \leq \lambda$ and $\lambda_{min} = -y_{min}.b_1$.

$+\lambda \leq y.b_2$ and $\lambda_{max} = +y_{max}.b_2$.

Proof. The first Point of the duality theorem indicates that : $y.b \geq c.x$ where, for the problem (3.10), $b_1^t.y^t = \begin{pmatrix} -L^t & -(T^-)^t & (T^+)^t \end{pmatrix}.y^t \geq \begin{pmatrix} 0 \ldots 0 & -1 \end{pmatrix} \begin{pmatrix} x \\ \lambda \end{pmatrix} = -\lambda.$

Moreover, the optimal values associated are equal (point 2) if the two problems admit a finit optimal solution. Similarly, we have

$b_2^t.y^t = \begin{pmatrix} U^t & (-T^-)^t & (T^+)^t \end{pmatrix}.y^t \geq \begin{pmatrix} 0 \ldots 0 & +1 \end{pmatrix} \begin{pmatrix} x \\ \lambda \end{pmatrix} = +\lambda.$ ■

3.5.3.3 Improvement 1

The study of the part [5] shows that the bounds of the cycle time depends on only the matrices which define the P-time events graph and not on the vectors L and U. As a consequence, we can simplify the approaches 1 and 2 if we consider only the optimisation of the cycle time ($Cr = \begin{pmatrix} 0 \ldots 0 & -1 \end{pmatrix}$), by taking $L = 0$ (resp., $U = 0$) . This hypothesis leads to a shift on the first dates of shooting of the transitions in the approach 1.

We denote $y = \begin{pmatrix} y_1 & y_2 \end{pmatrix}$ where y_1 has the dimension of x.

Property 4. *The problem (P) can be rewritten in the following form :*
min $\begin{pmatrix} -(T^-)^t & (T^+)^t \end{pmatrix} .y_2^t$
under :

$$\begin{cases} y_2 . \begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} \geq 0 \\ y_2 . \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix} = -1 \\ y_2 \geq 0 \end{cases} \tag{3.12}$$

Proof

As $L = 0$, only the subvector y_2 is optimized and we obtain $\min y. \begin{pmatrix} 0 & -(T^-)^t & (T^+)^t \end{pmatrix}^t$ under $y. \begin{pmatrix} -I & 0 \\ A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} = -Cr$ with $y \geq 0$. Recall that $Cr = \begin{pmatrix} 0 \ldots 0 & 1 \end{pmatrix}$.

The problem becomes :

$y_1 . \begin{pmatrix} -I & 0 \end{pmatrix} + y_2 . \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} = -Cr = \begin{pmatrix} 0 \ldots 0 & -1 \end{pmatrix}$.

So, we can deduce that $y_2 . \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix} = -1$ and $y_2 . \begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} \geq (0 \ldots 0)$ as $y_1 \geq 0$

The vector y_1 which is not optimized is obtained with $y_1 . = y_2 . \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} + Cr$. ■

Remark 10. *In the case of a P-time Event Graph, we can easily deduce the matrices by using the relations* $A_{11} = W = -A_{21}$ *and* $A_{12} = -M_0 = -A_{22}$.

3.5.3.4 Improvement 2

The following theorem will analyze the first relationship (3.12).

Theorem 10. *(Stiemke 1915, [16]). For a matrix A, the following cases are mutually exclusive from each other.*

- *case 1.* $Ax = 0$, $x > 0$ *has a solution* x.

– *case 2.* $y.A \geq 0$ *and* $y.A \neq 0$ *has a solution* y.

Property 5. *In relation (3.12) of the property 4,* $y_2.\begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} \geq 0$ *becomes* $y_2.\begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} = 0$ *if* $\begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix}.x = 0$, $x > 0$ *has a solution* x.

Proof

The application of the theorem of Stiemke entails that the case 2 has no solution : the system $y.A \geq 0$ and $y.A \neq 0$ has no solution y. On the other hand, the condition $y.A \geq 0$ in (3.12) must be checked in order to calculate the cycle time. The only possibility for the system is $y.A = 0$. ■

Remark 11. *In an event graph, there is* $x > 0$ *as* $W.x = 0$ *as each line of* W *has exactly a coefficient 1 and -1 (case 1). It can be conclude that the events graph is "consistent", ie there is a sequence of shooting and an initial marking which allows to find the same marking under the condition that each transition is fired at least once (see page 567 [14]). For a P-time Event Graph, we have the case 1 as* $A_{11} = W = -A_{21}$.

3.5.3.5 Example continued.

Approach 2

The Theorem of Stiemke can be applied as $x = \begin{pmatrix} 1 & 1 & 1 & 1 \end{pmatrix}^t$ is a solution to $\begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix}.x = 0$, $x > 0$. We can use the simplified dual approach 2 with the relation (3.12) with $y_2.\begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} = 0$.

Dual $\lambda_{\min}$
$y = \begin{pmatrix} 0 & 2.5 & 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$ with $f = 9.5$.
Dual $\lambda_{\max}$
$y = \begin{pmatrix} 0 & 0 & 2 & 0.33 & 0 & 0 & 1.66 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$ with $f = 13.66$. ■

3.6 Conclusion

In this chapter, we generalize the model of the P-time Event graph [10] by allowing the dependence between the time durations and considering its performance analysis. We first show that this new model can completely be described by a polyhedron $Ax \leq b$ which generalizes [5] [6]. Different to the matrices of a P-time Graph where there are correspondances between matrix A and incidence matrices, the matrices can have some entries which depend on time parameters and each row can contain more than two entries. Therefore, the system cannot be completely analyzed with a classical graph theory like in the case of a Timed Event Graph or a P-time Event Graph : we propose the use of linear programming in this chapter.

An objective of this chapter is the analysis of the performance with a criterion function of the cycle time but also of the the time durations associated with places : We rewrite the objective in the form of a linear programming problem whose resolution generates the optimal initial dates and cycle time of the problem. The direct application of these initial dates leads to a 1-periodic behavior of the system with the calculated cycle time.

Using the principle of duality, we also propose a second approach which can calculate the optimal cycle times. The main advantage is that the different results and conditions only depend on the constants of the system and not the dates of firing. We show in the example that the cycle time can be associated with structures which are more general than circuits in the associated graph.

The application of the Lemma of Farkas gives conditions of consistency of the problem and an interval limiting the cycle time : this result shows that the concept of cycle time holds on the new model as the conditions depend on the characteristics of the system only. Finally, the application of the theorem of Stiemke allows a simplification under an algebraic condition. This last one is satisfied if the study is limited to P-time Event graphs but also on the illustrative example which contains a dependence between different time durations.

3.7 Appendix

Now let us express system inequalities (3.1) on a reduced horizon. Such a form will simplify the calculations. The objective is to establish an equivalent model such that each place of the graph contains only zero or one token initially.

As a place contains a maximum number of m tokens, the general idea is to split each place containing m tokens into m places, where each place contains only one token. A systematic procedure is as follows.

Let us introduce new variable X , that is :

$X(k) = \begin{pmatrix} X_0(k) & .... & X_i(k) & .... & X_{m-1}(k) \end{pmatrix}$ with $X_i(k) = x(k-m+i+1)$. By construction, we have : $X_{m-1}(k) = x(k)$ and $X_i(k) = X_{i+1}(k-1)$ for i going from 0 to $m-2$.

So, system (3.1) becomes :

$$\begin{pmatrix} G'_1 & G'_0 \end{pmatrix} \times \begin{pmatrix} X(k-1) \\ X(k) \end{pmatrix} \leq (-T)$$

where : $G'_1 = \begin{pmatrix} G_m & 0 & \dots & \dots & 0 \end{pmatrix}$

and $G'_0 = \begin{pmatrix} G_{m-1} & G_{m-2} & \dots & G_1 & G_0 \end{pmatrix}$.

By completing the system with following $(m-1) \times |TR|$ relations,

$$X_i(k) - X_{i+1}(k-1) \leq 0$$

for $i = 0$ to $m-2$. The system can be written as follows :

$$\begin{pmatrix} N_{01}^{-} & N_{00}^{-} \\ N_{01}^{+} & N_{00}^{+} \end{pmatrix} \times \begin{pmatrix} X(k-1) \\ X(k) \end{pmatrix} \leq \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

where matrix $N_{01}^{-} = -N_{01}^{+}$ of dimension $((m-1) \times |TR| \times m)$ is an subdiagonal of identity matrices immediately above the main diagonal, while the matrix $N_{00}^{-} = -N_{00}^{+}$ is a diagonal of negative identity matrices.

Finally, the concatenation of the two systems gives the algebraic form :

$$\begin{pmatrix} H^{-} \\ H^{+} \end{pmatrix} . \begin{pmatrix} X(k-1) \\ X(k) \end{pmatrix} \leq \begin{pmatrix} -T^{-} \\ 0 \\ T^{+} \\ 0 \end{pmatrix}$$

where $H^{-} = \begin{pmatrix} G_1^{\prime -} & G_0^{\prime -} \\ N_{01}^{-} & N_{00}^{-} \end{pmatrix}$ et $H^{+} = \begin{pmatrix} G_1^{\prime +} & G_0^{\prime +} \\ N_{01}^{+} & N_{00}^{+} \end{pmatrix}$.

References

[1] F. Baccelli, G. Cohen, G.J. Olsder J.P. and Quadrat, Synchronization and Linearity. An Algebra for Discrete Event Systems, available from http ://maxplus.org, New York, Wiley, 1992.

[2] S.M. Burns, Performance analysis and optimization of asychronous circuits, Ph.D., Institute of Technology, Pasadena, California, USA.

[3] J. Campos, G. Chiola, J.M. Colom and M. Silva. Properties and Performance Bounds for Timed Marked Graphs. IEEE Trans. on Circuits and Systems. Vol. 39,N°5, may 1992.

[4] J-C. Culioli. Introduction à l'optimisation, Ellipses 1994.

[5] Ph. Declerck, A. Guezzi and J.-L. Boimond. Cycle Time of P-time Event Graphs, 4th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2007), Angers, France, 09-12 may 2007. Avalaible from http ://www.istia.univ-angers.fr/˜declerck/recherche.html

[6] Ph. Declerck, A. Guezzi, Cécile Gros, Temps de cycle des Graphes d'Événements Temporisés et P-temporels, CIFA 08, Bucharest, Romania, September 2008.

[7] F. Defossez, S. Collart-Dutilleul and P. Bon, Temporal requirements checking in a safety analysis of railway systems, FORMS/FORMAT 2008, Symposium on Formal Methods for Automation and Safety in Railway and Automotive Systems, TU Braunschweig and Budapest University of Technology and Economics, October, 2008.

[8] A. Giua, A. Piccaluga and C. Seatzu. Optimal Token Allocation in Timed Cyclic Event-Graphs. Proc. 4th Workshop on Discrete Event Systems. pp. 209-218, August 2000.

[9] A. Guezzi, Ph. Declerck, J.-L. Boimond, Commande de graphes d'événements temporisés sur un horizon glissant, Special issue of Journal Européen des Systèmes Automatisés (JESA), 2009, vol. 43/7-9, pp. 1097-1111.

[10] W. Khansa. Réseaux de Petri P-temporels. Contributionà l'étude des systèmes à Evénements discrets. Thèse. Université de Savoie. Mars 1997.

[11] L. Libeaut, Sur l'utilisation des dioïdes pour la commande des systèmes à événements discrets, Ph.D. thesis, Université de Nantes et École Centrale de Nantes, Nantes, France, 1996.

[12] J. Magott. Performance Evaluation of Concurrent Systems using Petri Nets. Information Processing Letters 18 (1984) 7-13 North-Holland.

[13] MuDer Jeng. Comments on ”Timed Petri Nets in Modeling and Analysis of Cluster tools”. IEEE Trans. on Automation Science and Engineering. Vol.2, N°.1, january 2005.

[14] T. Murata. Petri Nets : Properties, Analysis and Applications, Proceedings of the IEEE, Vol. 77, No. 4, 1989.

[15] C.V. Ramamoorthy and S. Gary H0. Performance Evaluation of Asynchronous Concurrent Systems Using Petri Nets. IEEE Trans. on Software Engineering, Vol. SE-6, N°.5, September 1980.

[16] A. Schrijver. Theory of linear and integer programming. John Wiley and Sons, 1987.

Chapitre 4

Commande avec spécifications

Dans le cadre de l'analyse de performances, le chapitre précédent a montré qu'il était possible d'appliquer la programmation linéaire sans utiliser la théorie des graphes ou l'algèbre des dioïdes. Dans le mémoire de Abdelhak Guezzi [23] et différents travaux [22], nous avons également prouvé qu'il était possible, avec l'algèbre standard, de faire de la commande au moins dans le cas simple du graphe d'événements temporisé sans spécifications. La considération de problèmes plus complexes ouvre clairement des perspectives. Ci-dessous, nous retournons à l'algèbre (max, +).

Ce chapitre a pour objectif la commande optimale d'un Graphe d'Evénements Temporisé devant suivre des spécifications exprimées par un modèle d'intervalles qui sera présenté. Celui-ci permet de décrire différentes classes de graphes d'événements temporels comme le graphe d'événements P-temporel et le graphe d'événements à flux. Le problème est reformulé sous une forme du type point fixe, la fonction f étant du type (min, max, +). La théorie spectrale permet de donner des conditions d'existence d'une solution. La question de l'équivalence de ces résultats dans l'algèbre conventionnelle est une question encore à traiter.

4.1 Introduction

In this chapter, we focus on the control problem of Timed Event Graphs defined as follows. In a Timed Event Graph, some events are stated as controllable, meaning that the corresponding transitions (input) may be delayed from firing until some arbitrary time provided by a supervisor. Moreover, we assume that the Timed Event Graph must follow specifications defined by a second model. As the specifications can be incoherent or too restrictive with respect to the Timed Event Graph, a first problem is to determine whether there exist control actions which will restrict the system to that behavior. If the trajectories of the Timed Event Graph can follow additional specifications, a second objective is to determine the greatest input in order to obtain the desired behavior defined by the static constraints (the desired output) and the dynamic constraints (expressed by a second model).

In this chapter, this second model is defined by a set of Time Interval Systems, for which the time evolution is not strictly deterministic. Introducing lower and upper bounds on the dates of firing of all the transitions, it is described by intervals which use the operations of maximization, minimization and addition to define the lower and upper bound constraints. P-time Event Graphs and Time Stream Event Graphs can algebraically be modeled by interval systems.

In the sequel, two classes of specifications are considered :

– Time Interval Systems where the lower bound is a (max, +) expression
– P-time Event Graphs in the next chapter.

In this chapter, we consider that each transition is observable : the event date of each transition firing is assumed to be available. No hypothesis is taken on the structure of the Event Graphs which does not need to be strongly connected. The initial marking should only satisfy the classical liveness condition and the usual hypothesis that places should be First In First Out (FIFO) is taken.

The chapter is structured as follows : The different models and the interval (min, max, +) systems are described in the following part. Based on a fixed point approach, we successively present the optimal control problem and analyze the existence of a finite solution through spectral theory. We then present an algorithm which determines the greatest control. Finally, the approach is applied to a general example. Let us recall that the notations are given in the chapter ""Vivacité" : maximization, minimization and addition operations are denoted respectively $\oplus$, $\wedge$ and $\otimes$; $\varepsilon = -\infty$ and $T = +\infty$.

4.2 Pedagogical example

Example. A simple real-world problem : education system

Plant. Described by the Timed Event Graph in Figure 4.1, the plant corresponds to the following work of a professor : the lesson is composed of a lecture (duration : T_1) followed by practical work (duration : T_2). Transitions express the following events. x_1 : beginning of the lecture ; x_2 : beginning of practical work ; x_3 : end of practical work ; u_1 : decision to start the lecture ; u_2 : decision to start practical work.

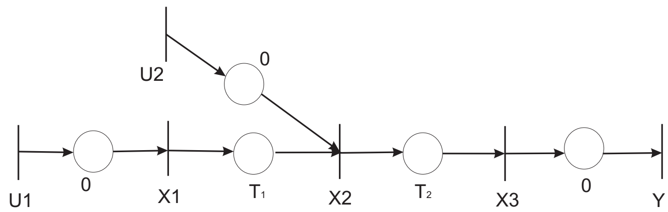

FIG. 4.1 : Timed Event Graph (plant : tasks of the professor)

Moreover, the teacher must follow the official instructions : a lesson must not exceed T_3^+ and not be less than T_3^-. These specifications can be described by a new Event Graph which can be a simple P-time Event Graph. So, $T_3^- \otimes x_1(k) \leq x_3(k) \leq T_3^+ \otimes x_1(k)$.

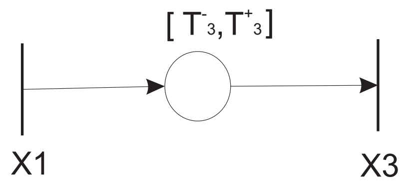

FIG. 4.2 : Time Event Graph (specification 1 : official instructions)

Problem. The lesson must stop before the daily closing time of school which corresponds to the desired output z. A problem can be the determination of the latest times to begin the lesson such that each specification is satisfied. If the teacher begins the lesson after this date which corresponds to the greatest control u, the practical work cannot be finished ($y \not\leq z$).

The lesson must now stop before the daily closing time of school from Monday (first day : $k = 1$) to Friday ($k = 5$) : Assume that desired output sequence $z(k)$ from $k = 1$ to 5 is 1140, 2580, 3600, 5460, 6480. So, $k_s = 0$ and $k_f = 5$. $T_1 = 60; T_2 = 90; T_3^- = 165; T_3^+ = 175$. The corresponding output sequence is as follows.

k	1	2	3	4	5
u_1	975	2415	3435	5295	6315
u_2	1050	2490	3510	5370	6390

■

The following subsystem can be deduced.

$$\begin{cases} x_1(1) \leq -T_1 + x_2(1) \\ x_2(1) \leq -T_2 + x_3(1) \\ x_3(1) \leq +T_3^+ + x_1(1) \end{cases} \tag{4.1}$$

Therefore, $x_1(1) \leq -T_1 - T_2 + T_3^+ + x_1(1)$. If $T_1 = 60$, $T_2 = 90$ and $T_3^+ = 175$, the inequality becomes $x_1(1) \leq -60 - 90 + 175 + x_1(1) = 25 + x_1(1)$ which is consistent.

Now, if we take $T_3^- = 135$ and $T_3^+ = 145$, we obtain $-T_1 - T_2 + T_3^+ = -5 < 0$. An incoherency appears as inequality $x_1(1) \leq -T_1 - T_2 + T_3^+ + x_1(1)$ gives $x_1(1) \leq -5 + x_1(1)$: the interpretation is that the lesson time of the professor is inconsistent with the official instructions.

But, we can add more complex specifications. The first student is present at the lecture and the practical work (duration $[T_4^-, T_4^+]$). The second student is only present at the practical work (duration $[T_5^-, T_5^+]$). Moreover, the lesson stops with the departure of the last student : when the students switch off, they go out (specification 2). This new specification can be described by a Time Stream Event Graph (specification 2 : two students)

$T_4^- \otimes x_1(k) \oplus T_5^- \otimes x_2(k) \leq x_3(k) \leq T_4^+ \otimes x_1(k) \oplus T_5^+ \otimes x_2(k)$

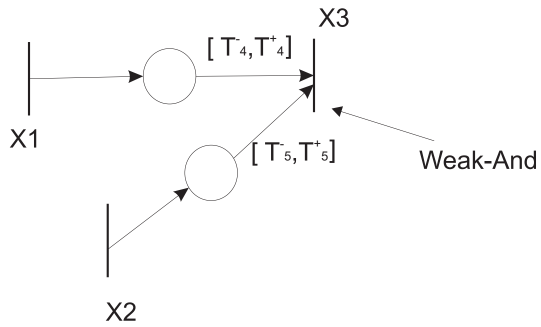

FIG. 4.3 : Time Stream Event Graph (specification 2 : two students)

4.3 Specifications defined by Time Interval Sytems

In this chapter, we assume that a set of Time Interval models describes the specifications. Let us give more details on this class of models.

Discrete event dynamic systems involving synchronization can be modeled by several types of Petri nets. The class of Time Event Graphs includes P-time Event Graphs and Time Stream Event Graphs which both extend the application field of Timed Event Graphs. For example, Time Stream Event Graphs allow the specification of synchronization requirements of multimedia applications [18] and the description of complex synchronizations. The Time Stream Petri nets present different types of semantic rules such as “And", “Weak-And", “Strong-Or", “Or", “Master" and their variations [19], which correspond to different temporal evolutions. We below show that Timed Event Graphs, P-time Event Graphs and Time Stream Event Graphs can be modeled by a new class of systems called interval systems, for which the time evolution is not strictly deterministic, but rather belongs to intervals which use the operations of maximization, minimization and addition to define the lower and upper bound constraints. Therefore, we will express the interval of firing of each transition for different Time Event Graphs.

4.4 Time Interval models

Let us recall that Event Graphs constitute a subclass of Petri nets in which each place has exactly one upstream and one downstream transition.

Alleviating the presentation, the following assumption is made : every internal place contains at the most one token ; every input (respectively, output) place contains no token : this form can simply be obtained, by elementary transformations such as duplication of places or by algebraic transformations.

4.4.0.6 Timed Event Graphs

It is well known that the model of a Timed Event Graph in the (max, +) dioid (see [1] for details), is as follows :

$$x(k) \geq A_1^- \otimes x(k-1) \oplus A_0^- \otimes x(k) \oplus B \otimes u(k) \tag{4.2}$$

with $(A)_{ij}, (B)_{ij}, \in \mathbb{R}_{\max}$

This model represents nondecreasing trajectories if matrix A is simply modified such that $x(k) \geq x(k-1)$. This remark holds for every model considered in this paper.

An equivalent model can also be obtained if the lower bound constraint defined by f^- is max-only ($f^-(x(k-1), x(k), u(k)) = A_1^- \otimes x(k-1) \oplus A_0^- \otimes x(k) \oplus B \otimes u(k)$) and the upper bound constraint $f^+()$ is infinite.

$$\begin{cases} x(k) \geq f^-(x(k-1), x(k), u(k)) = A_1^- \otimes x(k-1) \oplus A_0^- \otimes x(k) \oplus B \otimes u(k) \\ x(k) \leq f^+(x(k-1), x(k), u(k)) = +\infty \end{cases} \tag{4.3}$$

Consequently, a Timed Event Graph can also be described by an interval model .

The usual assumption that there is no extra delay for firing transitions whenever tokens are all available, can be made and leads to the following equality.

$$x(k) = A_1^- \otimes x(k-1) \oplus A_0^- \otimes x(k) \oplus B \otimes u(k) \tag{4.4}$$

This equation also leads to another interval system whose bounds are equal and defined by :

$$\begin{cases} x(k) \geq f^-(x(k-1), x(k), u(k)) = A_1^- \otimes x(k-1) \oplus A_0^- \otimes x(k) \oplus B \otimes u(k) \\ x(k) \leq f^+(x(k-1), x(k), u(k)) = A_1^- \otimes x(k-1) \oplus A_0^- \otimes x(k) \oplus B \otimes u(k) \end{cases} \tag{4.5}$$

4.4.0.7 P-time Event Graphs

In this model, time constraints of stay of the tokens is associated with each place. Coherent with dioid $\overline{\mathbb{R}}_{max}$, we associate with each place a temporal interval defined in $\mathbb{R}^+ \times (\mathbb{R}^+ \cup \{+\infty\})$.

Definition 6. ***(P-time Event Graph)*** *A P-time Event Graph is a pair $< R, IS >$ where R is an Event Graph*

$IS : P \longrightarrow \mathbb{R}^+ \times (\mathbb{R}^+ \cup \{+\infty\})$

$p_i \longrightarrow IS_i = [a_i, b_i]$ *with* $0 \leq a_i \leq b_i$

IS_i is the static interval of residence time or duration of a token in place p_i belonging to the set of places P. The token must stay in place p_i during the minimum residence duration a_i. Before this duration, the token is in state of unavailability to firing transition t_j. Value b_i is a maximum residence duration after which the token must thus leave place p_i. If not, the system is found in a token-dead state. So, the token is available to fire transition t_j in the interval time $[a_i, b_i]$. Let S_i denote the set of input transitions of transition i. The system is as follows. $\bigoplus_{j \in S_i} (x_j(k-m_j)+a_j) \leq x_i(k) \leq \bigwedge_{j \in S_i} (x_j(k-m_j)+b_j)$ with m_j the initial marking of place p_j. The lower bound (respectively upper bound) is a (max, +) function (respectively (min, +) function) and consequently, a P-time Event Graph is an example of interval system whose type is ((max, +), (min, +)). If $m_j = 1$ for every internal place, this is defined by

$$\begin{cases} x(k) \geq f^-(x(k-1), x(k), u(k)) = A_1^- \otimes x(k-1) \oplus A_0^- \otimes x(k) \oplus B^- u(k) \\ x(k) \leq f^+(x(k-1), x(k), u(k)) = A_1^+ \odot x(k-1) \wedge A_0^+ \odot x(k) \wedge B^+ \odot u(k) \end{cases} \tag{4.6}$$

Remark 12. *If every upper bound of places equals infinite, the model of a Timed Event Graph is obtained (4.2).*

4.4.0.8 Time Stream Event Graphs

Definition 7. ***(Time Stream Event Graph[11] [18])*** *Let I_j be a set of upstream arcs of a transition j and P_j be the corresponding set of upstream places. A Time Stream Event Graph is an Event-Graph such as :*
a) an interval $[\alpha_i, \beta_i] \in (\mathbb{R}^+ \cup \{0\}) \times (\mathbb{R}^+ \cup \{+\infty\})$ is associated with each $a_i \in I_j$ (usually defined on $\mathbb{Q}^+$, the limits of intervals are generalized to $\mathbb{R}^+$, which does not introduce new difficulties) ;
b) a special semantic rule of firing associated with each transition is defined below.

Considering one outgoing arc from a given place, when a token is received by that place at time x, the token should remain in the place during an amount of time defined by a value within the range $[x+\alpha, x+\beta]$ associated with the arc. As the firing time of a transition which has more than one input arc, depends on the nature of the processes which will be synchronized, different semantic rules of firing may be associated with a transition.

Definition 8. *For a transition i, let I_i denote a set of upstream arcs and P_i the corresponding set of upstream places. A transition i of the type "And" and "Weak-And" is fired at time x_i if and only if the two following conditions are satisfied :*

a) transition i is enabled for the current marking : every upstream place j of P_i contains at least one token. Let x_j be the entrance date of the token which is also the date of firing of the upstream transition of this place.

b) For the **semantic rule And,** the value of x_i is as follows : $(x_j + \alpha_j) \leq x_i \leq (x_j + \beta_j)$ for every upstream place $p_j \in P_i$ and arc $a_j \in I_i$ (every time condition has to be fulfilled). The model can be written as follows : If m_j the initial marking of the place p_j, the following expression can be written for each transition,

$\bigoplus_{j \in P_i} (x_j(k - m_j) + \alpha_j) \leq x_i(k) \leq \bigwedge_{j \in P_i} (x_j(k - m_j) + \beta_j)$ if the semantic rule is And ;

Let us notice that the inequalities of P-time Event Graph correspond to semantic rule And of Time Stream Event Graph.

For the **semantic rule Weak-And,** the value of x_i is as follows : $(x_i + \alpha_i) \leq x_i$ for every upstream place $p_j \in P_i$ and arc $a_j \in I_i$ and $\exists j \in P_i,\ x_i \leq (x_j + \beta_j)$ (the firing may wait until the last time interval).

$\bigoplus\limits_{j \in P_i} (x_j(k - m_j) + \alpha_j) \leq x_i(k) \leq \bigoplus\limits_{j \in P_i} (x_j(k - m_j) + \beta_j)$ if the semantic rule is Weak-And.

Therefore, for these two semantic rules, the upper bound is either min-only or max-only. Now, we will describe the Time Stream Event Graph in the form of an interval system. The following definitions allow the introduction of a standard description.

Let $f \in F(n, n)$. A subset $S \subset F(n, n)$ is said to be a max-representation of f if S is a finite set of (max,+) functions such that $f = \bigwedge\limits_{g \in S} g$. A subset $R \subset F(n, n)$ is said to be a min-representation of f if R is a finite set of (min,+) functions such that $f = \bigwedge\limits_{g \in R} g$. The mutual distributivity of $\otimes$ and $\wedge$ in the scalar case ($(x \oplus y) \wedge z = (x \wedge z) \oplus (y \wedge z)$ and $(x \wedge y) \oplus z = (x \oplus z) \wedge (y \oplus z)$) entails every (min, max, $+$) function having a max-representation and a min-representation. In short, any Time Stream Event Graph for semantic rules And and Weak-And, can be modeled under the following general form whose type is ((max, $+$), (min, max, $+$)) :

$$\begin{cases} x(k) \geq f^-(x(k-1), x(k), u(k)) = A_1^- \otimes x(k-1) \oplus A_0^- \otimes x(k) \oplus B^- u(k) \text{ and} \\ x(k) \leq f^+(x(k-1), x(k), u(k)) = \bigwedge\limits_{i=1}^{i_{\max}} g_i(x(k-1), x(k), u(k)) \text{ where} \\ g_i(x(k-1), x(k), u(k)) = A_{1,i}^+ \otimes x(k-1) \oplus A_{0,i}^+ \otimes x(k) \oplus B_i^+ \otimes u(k) \end{cases} \tag{4.7}$$

The finite temporizations of $A_1^-, A_0^-, B^-, A_{1,i}^+, A_{0,i}^+$ and B_i^+ belong to $\mathbb{R}^+$ and $i_{\max}$ is the maximal number of (max,+) terms in function f^+.

The general form (4.7) includes the state equation of Timed Event Graphs (equation 4.4) because the relevant interval model is found by taking : $i_{\max} = 1$. We deduce $A_1^- = A_1^+ = A_1$, $A_0^- = A_0^+ = A_0$ and $B^- = B^+ = B$. Consequently, the algebraic model of Timed Event Graphs may be seen as a particular case of the interval model. Moreover, the case where each function $g_i(x(k-1), x(k), u(k))$ is simple, defines a subclass which includes the model of P-time Event Graphs [13]. Equally, let us notice that the inequalities of P-time Event Graph correspond to semantic rule And of Time Stream Event Graph.

Remark 13. *Functions g_i in the upper term $f^+(x(k-1), x(k), u(k))$ must satisfy a nondegeneracy condition which we shall describe in this part. The concatenation of $A_{1,i}^+$, $A_{0,i}^+$ and B_i^+, is denoted M_i and the notation $M_i(j, .)$ represents the line j of matrix M_i, and $M(., l)$ represents the column l of matrix M_i. For every row j, $M_i(j, .)$ must be different from ε or, in other words, each row of M_i must describe at least one connection with an input or an incoming internal transition. Some elements $M_i(j, l)$ of row j in $A_{1,i}^+$ or $A_{0,i}^+$ naturally equal ε (respectively in B_i^+) if there are no places connecting internal transition directly from x_l to x_j (from input transition u_l to internal transition x_j respectively). When row $M_i(j, .)$ does not describe at least one connection, then it becomes neutral if $M_i(j, .) = T$.*

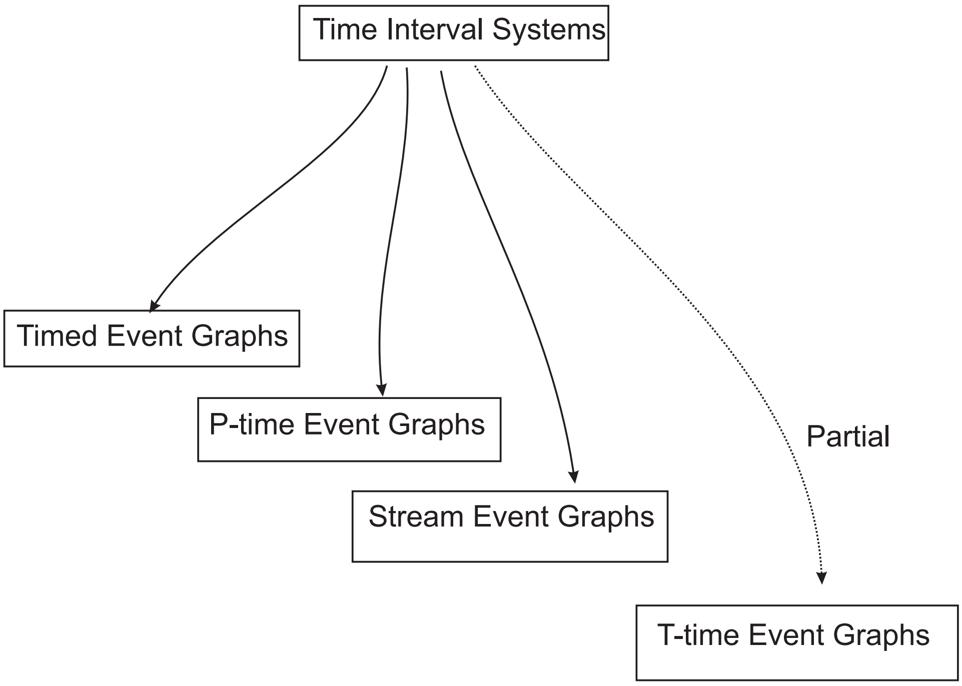

FIG. 4.4 : Time Interval Systems

In brief, the interval model (4.10) whose type is ((max, +), (min, max, +)), can describe Timed Event Graphs, P-time Event Graphs and Time Stream Event Graphs for the semantics And and Weak-And. Note that the interval model cannot completely describe a T-time Event Graph as this last one contains also a logical condition of the transition firing which cannot be easily rewritten (see the Ph.D. of Abdelhak Guezzi [23]).

4.5 Control synthesis

4.5.1 (min, max, +) algebraic models

Definition 9. *[37] A (min, max, +) function of type $(n, 1)$ is any function $f : \mathbb{R}^n \longrightarrow \mathbb{R}^1$, which can be written as a term in the following grammar : $f = x_1, x_2, \ldots, x_n \mid f \otimes a \mid f \wedge f \mid f \oplus f$ where a is an arbitrary real number ($a \in \mathbb{R}$).*

The vertical bars separate the different ways in which terms can be recursively constructed. A (min, max, +) function of type (n, m) is any function $f : \mathbb{R}^n \longrightarrow \mathbb{R}^m$, such that each component f_i is a (min, max, +) function of type $(n, 1)$. The set of functions (min, max, +) of the type (n, m) is denoted $F(n, m)$.

The evolution of the system is described by the following model, called an "interval model" or an "interval descriptor system", where f^- and f^+ are (min, max, +) functions.

$$f^-(x(k-1), x(k), u(k)) \leq x(k) \leq f^+(x(k-1), x(k), u(k)) \tag{4.8}$$

Vectors x and u are the state and the input, respectively. Variable $x_i(k)$ (respectively, $u_i(k)$) is the date of the k^{th} firing of transition x_i (respectively, u_i) for $k \in \mathbb{Z}$. In the same way and without reduction of generality, we can also introduce output y by

$$y(k) = C \otimes x(k) \text{ with } C_{ij} \in \mathbb{R}_{\max}$$

As the type of the system is defined by the types of functions f^- and f^+, we can characterize the model by the following pair (type of f^-, type of f^+) which defines different types of systems. Type ((min, max, $+$), (min, max, $+$)) naturally represents the more general mathematical case.

Let $f \in F(n,1)$. If f can be represented by a term that does not use $\wedge$, it is said to be max-only or (max, $+$). If f can be represented by a term that does not use $\oplus$, it is said to be min-only or (min, $+$). If f is both max-only and min-only, it is said to be simple. In this paper, the following assumption is made : for each interval model, the lower bound denoted f^- is max-only. Therefore, $f^-(x(k-1), x(k), u(k)) = A_1^- \otimes x(k-1) \oplus A_0^- \otimes x(k) \oplus B^- u(k)$. Model (4.8) with this assumption will be considered in the sequel.

4.5.2 Problem

Let us assume that an interval model describes the specifications of a Timed Event Graph. A first aim is the determination of an arbitrary trajectory following these models. Moreover, we may be given a sequence of dates at which one would like to see events occur at the latest, and we are asked to provide the latest input dates that would meet that objective. Therefore, the goal is to obtain the greatest control of a Timed Event Graph when the state and control trajectories are constrained by additional specifications. This problem is the generalization of the classical Just-In-Time control of Timed Event graphs where the ”Backward” equations express the optimal control (part 5.6 in [1], [8]). In other words, the "Just-in-time" objective is to calculate the greatest control u such that $y \leq z$ with $y(k) = C \otimes x(k)$ knowing :

a) The trajectory of the desired output z ;

b) The model of the plant described by a Timed Event Graph

$$x(k) = A_1 \otimes x(k-1) \oplus A_0 \otimes x(k) \oplus B \otimes u(k) \tag{4.9}$$

c) The model of additional specifications expressed by an interval system (4.8)

$$f^-(x(k-1), x(k), u(k)) \leq x(k) \leq f^+(x(k-1), x(k), u(k)) \tag{4.10}$$

where $f^-(x(k-1), x(k), u(k)) = A_1^- \otimes x(k-1) \oplus A_0^- \otimes x(k) \oplus B^- u(k)$ and f^+ is a (min, max, $+$) function.

The problem is defined on a horizon denoted $[\, k_s,\, k_f]$ ('s' for 'start', 'f' for 'final').

4.5.3 Example continued(education system)

$x(k) = A_1 \otimes x(k-1) \oplus A_0 \otimes x(k) \oplus B \otimes u(k)$
$y(k) = C \otimes x(k)$

$$A_1 = \varepsilon \ , \ A_0 = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon \\ T_1 & \varepsilon & \varepsilon \\ \varepsilon & T_2 & \varepsilon \end{pmatrix} , \ B = \begin{pmatrix} 0 & \varepsilon \\ \varepsilon & 0 \\ \varepsilon & \varepsilon \end{pmatrix} \text{ and } C = \begin{pmatrix} \varepsilon & \varepsilon & 0 \end{pmatrix}$$

P-Time Event Graph (specification 1 : official instructions)

$$\begin{cases} x(k) \geq A_1^- \otimes x(k-1) \oplus A_0^- \otimes x(k) \oplus B^- u(k) \\ x(k) \leq f^+(x(k-1), x(k), u(k)) \end{cases}$$

$$A_1^- = \varepsilon \ , \ A_0^- = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ T_3^- & \varepsilon & \varepsilon \end{pmatrix} , \ B^- = \varepsilon \text{ and } f^+(x(k-1), x(k), u(k)) = \begin{pmatrix} T \\ T \\ T_3^+ \otimes x_1(k) \end{pmatrix}$$

Time Stream Event Graph (specification 2 : two students)

$$\begin{cases} x(k) \geq A_1^- \otimes x(k-1) \oplus A_0^- \otimes x(k) \oplus B^- u(k) \\ x(k) \leq f^+(x(k-1), x(k), u(k)) \end{cases}$$

$$A_1^- = \varepsilon \ , \ A_0^- = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ T_4^- & T_5^- & \varepsilon \end{pmatrix} , \ B^- = \varepsilon \text{ and}$$

$$f^+(x(k-1), x(k), u(k)) = \begin{pmatrix} T \\ T \\ T_4^+ \otimes x_1(k) \oplus T_5^+ \otimes x_2(k) \end{pmatrix}$$

Interval system (specifications 1 and 2)

The following interval system describes specifications 1 and 2 and the relevant Event Graphs.

$$\begin{cases} x(k) \geq A_1^- \otimes x(k-1) \oplus A_0^- \otimes x(k) \oplus B^- u(k) \\ x(k) \leq f^+(x(k-1), x(k), u(k)) \end{cases}$$

$$A_1^- = \varepsilon \ , \ A_0^- = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ T_3^- \oplus T_4^- & T_5^- & \varepsilon \end{pmatrix} , \ B^- = \varepsilon \text{ and}$$

$f^+(x(k-1), x(k), u(k)) = \begin{pmatrix} T \\ T \\ T_3^+ \otimes x_1(k) \wedge T_4^+ \otimes x_1(k) \oplus T_5^+ \otimes x_2(k) \end{pmatrix}$ which is a (min, max, $+$) function.

The following inequalities can be obtained from the different Event graphs. This set corresponds to subsystem 1 in the following parts.

$$\begin{cases} x_1 \leq -T_1 + x2 \\ x_2 \leq -\max(T_2, T_5^-) + x_3 \\ x_3 \leq +T_3^+ + x1 \end{cases}$$

Therefore, $x_1 \leq -T_1 + -\max(T_2, T_5^-) + T_3^+ + x1$. If $T_1 = 60$, $T_2 = 90$, $T_3^+ = 175$ and $T_5^- = 0$, the inequality becomes $x_1 \leq -60 + -\max(90, 0) + 175 + x1 = 25 + x1$ which is consistent. Now, if $T_3^+ = 145$, an incoherency appears as the inequality gives $x_1 \leq -5 + x1$. The interpretation is that the lesson time of the professor is inconsistent with the official instructions.

4.6 Fixed point approach

4.6.1 Fixed point formulation

The problem is now reformulated as a fixed point problem.

Theorem 11. *For a Timed Event Graph (4.9) following an interval model (4.10), the problem of the greatest control can be written as follows : from k_s to k_f, search the greatest state and control of the following inequality $V \leq h(V)$ with* $V = \begin{pmatrix} x(k-1) \\ x(k) \\ x(k+1) \\ u(k) \end{pmatrix}$

$$h(V) = \begin{pmatrix} x(k-1) \\ h_{bw}(x(k+1)) \wedge h_{st}(x(k)) \wedge h_{fw}(x(k-1), x(k), u(k)) \\ x(k+1) \\ (B \oplus B^-)\backslash x(k) \end{pmatrix} \tag{4.11}$$

, $h_{bw}(x(k+1)) = (A_1 \oplus A_1^-)\backslash x(k+1)$ *,* $h_{st}(x(k)) = (A_0 \oplus A_0^-)\backslash x(k) \wedge (C\backslash z(k))$ *and* $h_{fw}(x(k-1), x(k), u(k)) = [A_1 \otimes x(k-1) \oplus A_0 \otimes x(k) \oplus B \otimes u(k)] \wedge f^+(x(k-1), x(k), u(k))$

Proof. The models are :

$f^-(x(k-1), x(k), u(k)) \leq x(k) \leq f^+(x(k-1), x(k), u(k))$ where $f^-(x(k-1), x(k), u(k)) = A_1^- \otimes x(k-1) \oplus A_0^- \otimes x(k) \oplus B^- u(k)$ and $f^+(x(k-1), x(k), u(k))$ is a (min, max, +) function.

$x(k) = A_1 \otimes x(k-1) \oplus A_0 \otimes x(k) \oplus B \otimes u(k)$ or $\begin{cases} A_1 \otimes x(k-1) \oplus A_0 \otimes x(k) \oplus B \otimes u(k) \leq x(k) \\ x(k) \leq A_1 \otimes x(k-1) \oplus A_0 \otimes x(k) \oplus B \otimes u(k) \end{cases}$

Therefore, the problem can be written

$$\begin{cases} (A_1 \oplus A_1^-) \otimes x(k-1) \oplus (A_0 \oplus A_0^-) \otimes x(k) \oplus (B \oplus B^-) \otimes u(k) \leq x(k) \\ x(k) \leq [A_1 \otimes x(k-1) \oplus A_0 \otimes x(k) \oplus B \otimes u(k)] \wedge f^+(x(k-1), x(k), u(k)) \\ y(k) = C \otimes x(k) \\ y(k) \leq z(k) \end{cases}$$

Consequently,

$$\begin{cases} x(k) \leq (A_1 \oplus A_1^-)\backslash x(k+1) \wedge (A_0 \oplus A_0^-)\backslash x(k) \\ u(k) \leq (B \oplus B^-)\backslash x(k) \\ x(k) \leq [A_1 \otimes x(k-1) \oplus A_0 \otimes x(k) \oplus B \otimes u(k)] \wedge f^+(x(k-1), x(k), u(k)) \\ x(k) \leq C\backslash z(k) \end{cases}$$

and the fixed point formulation (4.11) is found.

Expression (4.11) presents a "Backward" part $h_{bw}(x(k{+}1)$ but also, a static part $h_{st}(x(k), u(k))$ and a forward part $h_{fw}(x(k-1), x(k), u(k))$. This fact increases the complexity of the problem and forbids the writing of simple equations on a short horizon such as the classical backward equations. Moreover, even if there is no function $f^{+}(x(k-1), x(k), u(k))$ in the specifications, h is a (min, max, $+$) function. In short, we must solve a fixed-point problem of type $x \leq f(x)$ (if x exists) over the horizon of the desired output z with complex backward and forward interconnections defined by (min, max, $+$) functions.

Example (education system continued)

$$h_{bw}(x(k+1)) = T \ , \ h_{st}(x(k)) = \begin{pmatrix} T_1 \backslash x_2(k) \wedge (T_3^- \oplus T_4^-) \backslash x_3(k) \\ (T_2 \oplus T_5^-) \backslash x_3(k) \\ z(k) \end{pmatrix},$$

$$h_{fw}(x(k-1), x(k), u(k)) = \begin{pmatrix} u_1(k) \\ T_1 \otimes x1(k) \oplus u_2(k) \\ T_2 \otimes x_2(k) \wedge T_3^+ \otimes x_1(k) \wedge (T_4^+ \otimes x_1(k) \oplus T_5^+ \otimes x_2(k)) \end{pmatrix} \text{ and}$$

$$(B \oplus B^-) \backslash x(k) = \begin{pmatrix} x_1(k) \\ x_2(k) \end{pmatrix}$$

$$h\,(V) = \begin{pmatrix} x(k-1) \\ T_1 \backslash x_2(k) \wedge (T_3^- \oplus T_4^-) \backslash x_3(k) \wedge u_1(k) \\ (T_2 \oplus T_5^-) \backslash x_3(k) \wedge (T_1 \otimes x_1(k) \oplus u_2(k)) \\ z(k) \wedge T_2 \otimes x_2(k) \wedge T_3^+ \otimes x_1(k) \wedge (T_4^+ \otimes x_1(k) \oplus T_5^+ \otimes x_2(k)) \\ x(k+1) \\ x_1(k) \\ x_2(k) \end{pmatrix}$$

4.6.2 Existence

The direct application of the famous theorem of Knaster and Tarski will a priori define the solution set .

Theorem 12. *[25] Let $(X, \leq)$ be a complete lattice and $f : X \to X$ be a monotone function. Let $Y = \{x \in X \mid f(x) = x\}$ be the set of fixed points of f. Then*

a) $\inf Y \in Y$ *, and* $\inf Y = \inf\{x \in X \mid f(x) \leq x\}$

b) $\sup Y \in Y$ *, and* $\sup Y = \sup\{x \in X \mid x \leq f(x)\}$

The dimension of V is denoted $dim(V)$.

Proposition 4.1. *Let $h\,(V) : \overline{\mathbb{R}}_{\max}^{dim(V)} \to \overline{\mathbb{R}}_{\max}^{dim(V)}$ be defined by (4.11). Let $Y = \{V \in \overline{\mathbb{R}}_{\max}^{dim(V)} \mid h\,(V) = V\}$ be the set of fixed points of f. Then*

a) $\inf Y \in Y$*, and* $\inf Y = \inf\{V \in \overline{\mathbb{R}}_{\max}^{dim(V)} \mid h\,(V) \leq V\}$

b) $\sup Y \in Y$*, and* $\sup Y = \sup\{V \in \overline{\mathbb{R}}_{\max}^{dim(V)} \mid V \leq h\,(V)\}$

Proof. The proof is immediate because $(\mathbb{R} \cup \{-\infty\} \cup \{+\infty\}, \leq)$ is a complete lattice and function $h(.)$ is isotone. If an algorithm gives the greatest solution of $x \leq f(x)$, this solution also satisfies the relevant equality. Function $h()$ is also discontinuous but Knaster and Tarski's theorem does not require continuity of the function.

Let us remark that the previous proposition does not lead to the existence of a finite solution because the considered lattice includes infinite elements. The aim of this part is to satisfy the existence of a finite solution in the control synthesis by using Theorem 13 presented below.

The property of homogeneity of functions (min, max, +) belonging to $F(n, m)$ is necessary to use the spectral vector below. This is defined as follows : $\forall \lambda \in \mathbb{R}, \forall x \in \mathbb{R}^n$ $f(\lambda \otimes x) = \lambda \otimes f(x)$ in the usual vector-scalar convention : $(\lambda \otimes x)_i = \lambda \otimes x_i$.

As function h contains the desired output z, h is not a (min, max, +) function of type (n, m) which is homogeneous. The form of our practical problem is to find the greatest x (if x exists) such that $x \leq f(x)$ with f a non-homogeneous (min, max, +) function which can be defined by the following grammar : $f = b, x_1, x_2, \ldots, x_n \mid f \otimes a \mid f \wedge f \mid f \oplus f$ where a, b are arbitrary real numbers ($a, b \in \mathbb{R}$). In the aim of applying the spectral theory about these functions, we will use a relaxation by associating a variable x_0 with b such that b is replaced by $b \otimes x_0$ in the above definition. So, the problem is to find the greatest $y = (x_0, x_1, ..., x_n)^t$ (if y exists) such that $x_0 = 0$ and $x_i \leq f_i(x_0, x_1, ..., x_n)^t)$ for $i \neq 0$. If we introduce the obvious inequality $x_0 \leq x_0$, the general problem becomes : find the greatest $y = (x_0, x_1, ..., x_n)^t$ (if y exists) such that $x_0 \leq x_0$ and $x_i \leq f_i(x_0, x_1, ..., x_n)^t)$ for $i \neq 0$ with $x_0 = 0$. In other terms, we have to solve the new system

$$y \leq F(y)$$

with F being a homogeneous function of type $(n+1, n+1)$. With each nonhomogeneous function f is associated a homogeneous function F, denoted with the same letters but in capitals.

Now, we recall some useful results of spectral theory.

Dynamics of the form are considered : $x(k) = f(x(k-1)), \forall k \geq 1$ and $x(0) = \xi \in \mathbb{R}^n$, where f is a (min, max, +) function of type (n, n) $\mathbb{R}^n \to \mathbb{R}^n$. The cycle time vector is defined by $\chi(f) = \lim_{k\to\infty} x(k)/k$ if it exists. It does not depend on ξ. The pair $(\eta, v) \in \mathbb{R}^2$ is an ultimately affine regime of f if there exists an integer K such that $\forall k \geqslant K$, $f(v + k\eta) = v + (k+1)\eta$. As a part of the paper is based on the following fundamental theorem, the corresponding proof gives a complete presentation of the approach. In the following theorem, the notions of cycle time and ultimately affine regime, which always exist in $F(n, n)$, make it possible to satisfy the existence of a solution of different inequalities and equalities.

Theorem 13. *[20] [8] [9] [21] Let f be a function of $F(n, n)$. The two following conditions are equivalent :*

(i) There is a finite x such that $x \leq f(x)$ (respectively, $x \geq f(x)$)

(ii) $\chi(f) \geq 0$ (respectively, $\chi(f) \leq 0$)

Proof

(i)⇒(ii) $x \leqslant f(x) \Rightarrow x \leqslant f(x) \leqslant f^2(x) \leqslant f^k(x)$ with $k \geqslant 1$ because f is nondecreasing. By definition, $\chi(f,x) = \lim_{k\to\infty} x(k)/k = \lim_{k\to\infty} f^k(x)/k \geqslant \lim_{k\to\infty} x/k = 0$ (x is finite).

(ii)⇒(i) In [9], it has been proved that any function in $F(n,n)$ has an ultimately affine regime and moreover, $\chi(f)$ equals η, for all ultimately affine regimes $(\eta, v) \in (R^n)^2$ of f. For an ultimately affine regime (η, v), we have $\forall k \geqslant K$, $f(v + k\eta) = v + (k+1)\eta$. We put $y = v + k\eta$ with $k \geqslant K$. The variable y verifies $f(y) = y + \eta$. As $\eta = \chi(f) \geqslant 0$, we deduce that it exists y such that $f(y) \geqslant y$.

Notation. Let us denote $X_l = (x(k)^t, x(k+1)^t, ..., x(k+l)^t)^t$ with horizon l and the dater algebraic function $\Phi_f(X_l)$ obtained by developing $f(x(k))$ algebraically on X_l. Therefore, $X_l \leq \Phi_f(X_l)$ describes every relation between the components of X_l. The following example illustrates this notation.

Example of notation

$f(x(k)) = \binom{ax_1(k)\oplus bx_2(k+1)}{cx_2(k)\oplus dx_1(k-1)}$ with $x(k) = \binom{x_1(k)}{x_2(k)}$

$X_2 = (x_1(k), x_2(k), x_1(k+1), x_2(k+1), x_1(k+2), x_2(k+2))^t$

$\Phi_f(X_2) = (ax_1(k) \oplus bx_2(k+1), cx_2(k),$

$ax_1(k+1) \oplus bx_2(k+2), cx_2(k+1) \oplus dx_1(k),$

$ax_1(k+2), cx_2(k+2) \oplus dx_1(k+1))^t$

Theorem 14. *Let us consider*

$$\begin{pmatrix} x_0 \\ V \end{pmatrix} \leq H \begin{pmatrix} x_0 \\ V \end{pmatrix} \tag{4.12}$$

with $H\begin{pmatrix} x_0 \\ V \end{pmatrix} = \begin{pmatrix} x_0 \\ x(k-1) \\ h_{bw}(x(k+1)) \wedge h_{st}(x_0, x(k)) \wedge h_{fw}(x(k-1), x(k), u(k)) \\ x(k+1) \\ (B \oplus B^-)\backslash x(k) \end{pmatrix}$

and $(x_0, x(k)^t, u(k)^t, x(k+1)^t, u(k+1)^t, ..., x(k+l)^t, u(k+l)^t)^t$. *There exists a finite solution satisfying system (4.7) on horizon* l *if and only if* $\chi(\Phi_H(X_l)) \geq 0$.

Proof. In $x(k) \leq h_{bw}(x(k+1)) \wedge h_{st}(x(k)) \wedge h_{fw}(x(k-1), x(k), u(k))$, constraint $x(k) \leq h_{st}(x(k)) = (A_0 \oplus A_0^-)\backslash x(k) \wedge (C\backslash z(k))$ becomes $x(k) \leq h_{st}(x_0, x(k)) = (A_0 \oplus A_0^-)\backslash x(k) \wedge (C\backslash(z(k) \otimes x_0))$. Consequently, the system defined by (4.11) can also be defined by system (4.12) under the condition $x_0 = 0$. The final inequality set presents the general form $x \leq F(x)$ which can be associated with the algebraic inequality $X_l \leq \Phi_F(X_l)$. As H is a homogeneous function, the spectral vector can be calculated $\chi(\Phi_H(X_l))$ and Theorem 13 applied. If the cycle time verifies the corresponding condition of existence, the interval system presents a solution. For any solution, an obvious translation can be applied in such a way that the first component equals zero. Existence of a solution entails the possibility of taking $x_0 = 0$ because of the homogeneity of H.

Remark 14. *Let us note that, as the left multiplication by a scalar is defined in the preliminaries, product* $z(k) \otimes x_0$ *can also be written* $x_0 \otimes z(k)$*.*

Before giving an example, the following preliminary results give a way of calculating the spectral vector.

Preliminary results

The calculation of the spectral vector can be done as follows. Denoted $G(A)$, an induced graph (or an associated graph) of a square matrix A is deduced from this matrix by associating a node i with column i and line i. For each $A_{ij} \neq \varepsilon$, an arc from node j towards node i is added. The following convention will be useful to (min, max, +) functions : all arcs derived from the same (max, +) row are bundled together with a line drawn through them near their arrow end [28] (see figure 4.5). The weight of a path p, $\mid p \mid_w$ is the sum of the labels on the edges of the path. The length of a path p, $\mid p \mid_l$ is the number of edges of the path. A circuit is a path which starts from and ends at the same node. If c is a circuit, its cycle average, denoted $m(c)$ is defined by $m(c) =\mid c \mid_w / \mid c \mid_l$ (The notation / represents the classical division). A node j is upstream from i, denoted $i \Longleftarrow j$, if either $i = j$ or if there is a path in $G(A)$ from j to i. A circuit c is upstream from i, denoted $i \Longleftarrow c$, if either $i \in c$ or if there is a path in $G(A)$ from a node $j \in c$ to i.Vector $\mu(A) \in \mathbb{R}^n$ is defined by $\mu_i(A) = \max\{m(c) \mid i \Longleftarrow c\}$. If $f \in F(n, n)$ is max-only and A is the associated matrix over $\mathbb{R}_{max}$, then $\chi(f)$ exists and $\chi(f) = \mu(A)$. The result is identical for min-only function.

Let $f \in F(n, n)$. Recall that a subset $S \subset F(n, n)$ is said to be a max-representation of f if S is a finite set of max-only functions such that $f = \bigwedge_{g \in S} g$. As each function of $F(n, n)$ is made up of n functions belonging to $F(n, 1)$, the set of (min, max, +) functions $F(n, n)$ has a natural representation as an n-fold Cartesian product : $F(n, n) = F(n, 1)$x...$F(n, 1)$. Let S_i be set $\{h_i$ such that $h_i \in S\}$. The rectangularization of S, denoted $rec(S)$, is defined by $rec(S) = S_1$xS_2x...xS_n. In other words, a set S of min-max functions is rectangular if for all h, $h' \in S$, and for all $i = 1, ..., n$, the function obtained by replacing the i-th component of h by the i-th component of h' belongs to S. Therefore, $rec(S)$ is finite when S is finite and $S \subset rec(S)$. If S and R are rectangular max and min-representations, respectively, of $f \in F(n, n)$, then $\chi(f) = \bigwedge_{h \in S} \chi(h) = \bigoplus_{g \in R} \chi(g)$.

Example (education system continued)

$rec(S) = \{G_{11}\}$x$\{G_{21}\}$x$\{G_{31}\}$x$\{G_{41}\}$x$\{G_{51}, G_{52}, G_{53}\}x\{G_{61}, G_{62}\}x\{G_{71}, G_{72}, G_{73}, G_{74}\}x\{G_{81}\}$

$G_{11}(V) = x_0$;

$G_{21}(V) = x_1(k-1)$;

$G_{31}(V) = x_2(k-1)$;

$G_{41}(V) = x_3(k-1)$;

$G_{51}(V) = T_1 \backslash x_2(k)$; $G_{52}(V) = (T_3^- \oplus T_4^-) \backslash x_3(k)$; $G_{53}(V) = u_1(k)$;

$G_{61}(V) = (T_2 \oplus T_5^-) \backslash x_3(k)$; $G_{62}(V) = T_1 \otimes x_1(k) \oplus u_2(k)$;

$G_{71}(V) = z(k) \otimes x_0$; $G_{72}(V) = T_2 \otimes x_2(k)$; $G_{73}(V) = T_3^+ \otimes x_1(k)$;$G_{74}(V) = (T_4^+ \otimes x_1(k) \oplus T_5^+ \otimes x_2(k))$;

$G_{81}(V) = x_1(k-1)$;

$G_{91}(V) = x_2(k-1)$;

$G_{10,1}(V) = x_3(k-1)$;

$G_{11,1}(V) = x_1(k)$;

$G_{12,1}(V) = x_2(k)$;

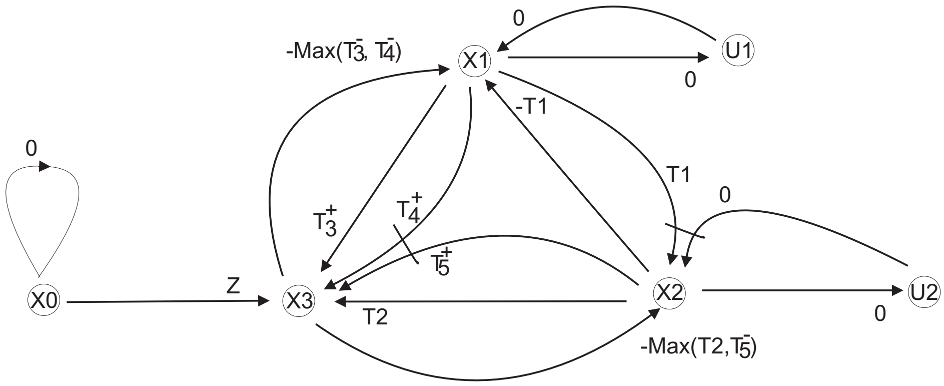

FIG. 4.5 : System

The max-only subsystems (3.2.4=24 possibilities) are produced by a combination of the previous functions. Their analysis leads to $\chi(\Phi_H(X_{+\infty})) = 0$ for the following values : $T_1 = 60; T_2 = 90; T_3^- = 165; T_3^+ = 175; T_4^- = 0; T_4^+ = 143; T_5^- = 0; T_5^+ = 98$. Below, the calculation is considered in two examples of subsystems. As the process is static, the functions $G_{21}(V), G_{31}(V), G_{41}(V), G_{81}(V), G_{91}(V)$ and $G_{10,1}(V)$ can be disregarded.

Subsystem 1

$$\begin{pmatrix} G_{11}(V) = x_0 \\ G_{51}(V) = T_1 \backslash x_2(k) \\ G_{61}(V) = (T_2 \oplus T_5^-) \backslash x_3(k) \\ G_{73}(V) = T_3^+ \otimes x_1(k) \\ G_{11,1}(V) = x_1(k) \\ G_{12,1}(V) = x_2(k) \end{pmatrix}$$

The substructure contains a unique circuit which must be positive or null. If $-T_1 - (T_2 \oplus T_5^-) + T_3^+ \geq 0$, the relevant spectral vector is positive or null.

Subsystem 2

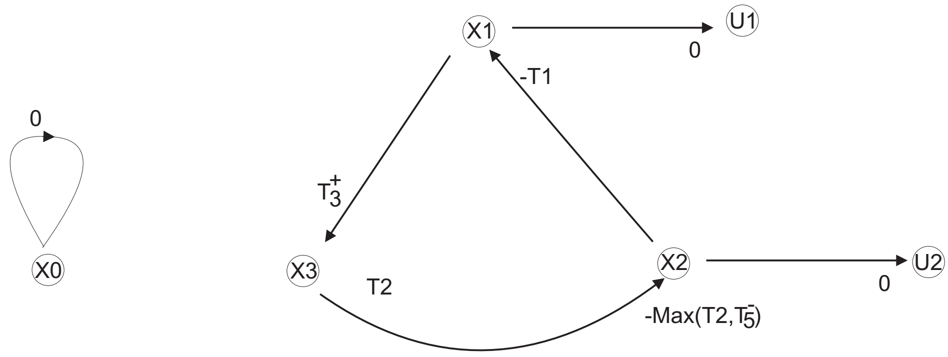

Fig. 4.6 : Subsystem 1

$$\left(\begin{array}{l} G_{11}(V) = x_0 \\ G_{52}(V) = (T_3^- \oplus T_4^-)\backslash x_3(k) \\ G_{62}(V) = T_1 \otimes x_1(k) \oplus u_2(k) \\ G_{72}(V) = T_2 \otimes x_2(k) \\ G_{11,1}(V) = x_1(k) \\ G_{12,1}(V) = x_2(k) \end{array}\right)$$

Functions G_{62} and $G_{12,1}$ constitute a null circuit and there is

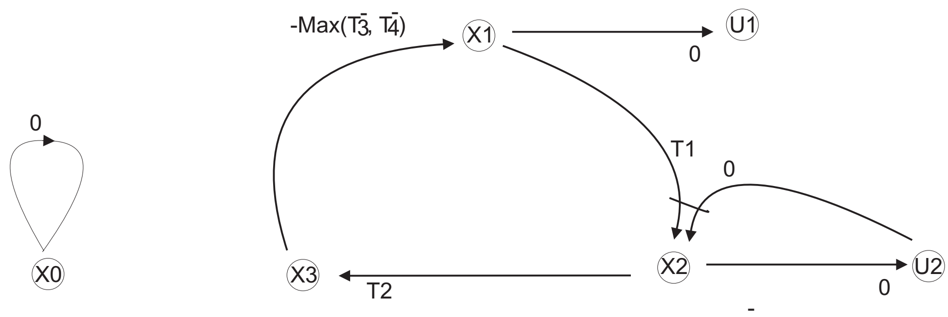

Fig. 4.7 : Subsystem 2

a path from x_2 to the other vertices. Even if the other circuit satisfies $-(T_3^- \oplus T_4^-) + T_1 + T_2 < 0$, the relevant spectral vector is null because the maximum vector is taken.

4.6.3 Structure

Now, the underlined structure of H, will be considered. Function H_1 defined below is similar to classical "Backward equations" for Timed Event Graphs. $\begin{pmatrix} x_0 \\ V \end{pmatrix} \leq H_1 \begin{pmatrix} x_0 \\ V \end{pmatrix}$ with

$$H_1 \begin{pmatrix} x_0 \\ V \end{pmatrix} = \left(\begin{array}{l} x_0 \\ x(k-1) \\ A_1\backslash x(k+1) \wedge A_0\backslash x(k) \wedge (C\backslash(z(k) \otimes x_0)) \\ x(k+1) \\ B\backslash x(k) \end{array}\right)$$

The structural observability (respectively controllability) gives a condition to observe an effect in the output (resp. transition) whose origin comes from one internal transition (resp. input) at least. It allows the introduction of the following propositions where an infinite horizon is considered.

Definition 10. *[1] An event graph is structurally controllable if, every internal transition can be reached by a path from one input transition at least. An event graph is structurally observable if, from every internal transition, there exists a path to one output transition at least.*

Proposition 4.2. *A structurally observable event graph defined by model (4.9) satisfies* $\chi(\Phi_{H_1}(X_{+\infty})) = 0$. *Moreover, the greatest solution denoted* β*, of the "Backward" part* $h_1(V) = V$ *satisfies* $h(\beta) \leq \beta$ *; The greatest solution of the "Backward" part gives a finite upper bound of the solutions of the inequality (4.11)* $V \leq h(V)$ *and the corresponding equality* $V = h(V)$.

Proof. The max-representation of $h(V)$ is $\bigwedge_{i=1}^{+\infty} h_i(V)$ with h_i a max-only function. The corresponding homogeneous function is $H\begin{pmatrix} x_0 \\ V \end{pmatrix} = \bigwedge_{i=1}^{+\infty} H_i \begin{pmatrix} x_0 \\ V \end{pmatrix}$.

- A circuit whose weight equals zero, is associated with x_0 because $x_0 \leq x_0$. Every known desired output value $z(k)$ defines an arc from vertex x_0 to a vertex associated with a component of the state for a given k. If we assume that the system is structurally observable, from every internal transition, there exists a path to one output transition in the Event Graph at least. In the associated graph, the direction of the paths are opposite and for every vertex corresponding to a transition at any k, there is a path going from x_0. Moreover, the same reasoning can be made for the inputs : as every input transition is connected to the system by definition, each vertex associated with an input for k can be reached by a path from one internal transition at least. Therefore, the definition of the spectral vector shows that $\chi(\Phi_{H_1}(X_{+\infty})) = 0$. Consequently, there is α such that $\alpha = H_1(\alpha)$. This solution verifies also, $H(\alpha) = \bigwedge_{i=1}^{+\infty} H_i(\alpha) \leq H_1(\alpha) = \alpha$. Particularly, the corresponding greatest solution of the "Backward" part $h_1(V) = V$ for $x_0 = 0$, denoted β verifies $h(\beta) \leq \beta$.

As the set of series is a complete lattice and h_1 is a monotone function, Theorem 12 can be applied : it proves that the greatest solution β of $h_1(V) = V$ is also the greatest solution satisfying,

$$V \leq h_1(V) \tag{4.13}$$

This result can also be proved by considering the set defined by (4.13) which is a complete lattice (Theorem 4.72 point 2 in [1]). Moreover, as (4.11) $V \leq h(V)$ is equivalent to the set of inequalities $V \leq h_i(V)$, each variable V must satisfy each inequality and particularly (4.13). Consequently, each solution V of (4.11) is less than or equal to β. Finally, this set $\{V \mid V \leq h(V)\}$ includes the solution set satisfying $V = h(V)$. The greatest solution of $V \leq h(V)\}$is denoted α.■

Remark 15. *Let us note that the reasoning does not need the assumption of structurally controllable system but only, the definition of input transition.*

Example (education system continued)

As the system is static, we take $V = \begin{pmatrix} x(k) \\ u(k) \end{pmatrix}$.

$A_1 \backslash x(k+1) \wedge A_0 \backslash x(k) \wedge (C \backslash (z(k))) = \begin{pmatrix} T_1 \backslash x_2(k) \\ T_2 \backslash x_3(k) \\ z(k) \end{pmatrix}$ and $B \backslash x(k) = \begin{pmatrix} x_1(k) \\ x_2(k) \end{pmatrix}$. If $z = 200$, $\beta = \begin{pmatrix} 50 \\ 110 \\ 200 \\ 50 \\ 110 \end{pmatrix}$ for $h_1(x(k)) = \begin{pmatrix} T_1 \backslash x_2(k) \\ T_2 \backslash x_3(k) \\ z(k) \\ x_1(k) \\ x_2(k) \end{pmatrix}$ and, using a classical fixed point approach (see part 4.7), $\alpha = \begin{pmatrix} 35 \\ 110 \\ 200 \\ 35 \\ 110 \end{pmatrix}$ for $h_{bw}(x(k+1)) \wedge h_{st}(x(k)) \wedge h_{fw}(x(k-1), x(k), u(k)) =$

$\begin{pmatrix} T_1 \backslash x_2(k) \wedge (T_3^- \oplus T_4^-) \backslash x_3(k) \wedge u_1(k) \\ (T_2 \oplus T_5^-) \backslash x_3(k) \wedge (T_1 \otimes x_1(k) \oplus u_2(k)) \\ z(k) \wedge T_2 \otimes x_2(k) \wedge T_3^+ \otimes x_1(k) \wedge (T_4^+ \otimes x_1(k) \oplus T_5^+ \otimes x_2(k)) \end{pmatrix}$. Therefore, the greatest control u is $\begin{pmatrix} 35 \\ 110 \end{pmatrix}$. The calculations give : $h_1(\beta) = \beta$; $h(\beta) \leq \beta$; $\alpha = h(\alpha)$; $\alpha \leq \beta$.■

Proposition 4.3. *In a structurally observable event graph, a solution satisfying the relevant equality of (4.11) $V \leq h(V)$ (respectively, of the relaxed form (4.12)) exists or, there is no solution in inequality (4.11) (respectively, (4.12))*

Proof. As a structurally observable event graph defined by model (4.9) satisfies $\chi(\Phi_{H_1}(X_{+\infty})) = 0$ (proposition 4.2), one can deduce that the spectral vector of the complete system is less than or equals to zero. The association with new terms can create strictly negative circuits which can entail that there is no solution.■

Example (education system continued)

As there is a path from node x_0 associated with a null loop ($x_0 \leq G_{11}(V) = x_0$) to each node, $\chi(\Phi_{H_1}(X_{+\infty})) = 0$. Now, if $T_3^+ = 145$, $-T_1 - (T_2 \oplus T_5^-) + T_3^+ = -5 \geq 0$ and some components of spectral vector $\chi(\Phi_H(X_{+\infty}))$ are negative. Therefore, $\chi(\Phi_H(X_{+\infty})) \not\geq 0$ and (4.11) $V \leq h(V)$ has no solution : the official instructions are too strict.

4.7 Algorithm

The effective calculation of the greatest control of inequality (4.11) can be made by a classical iterative algorithm of Mc Millan and Dill [26] or Walkup-Borriello [28] (see also, [24]) which particularizes the algorithm of Kleene to (min, max, +) expressions. The general resolution of $x \leq f(x)$ is given by the iterations of $x_{i+1} \leftarrow f(x_i) \wedge x_i$ if the finite starting point is greater than the final solution. Here, number i represents the number of iterations and not the number of components of vector x. Following this framework, we provide an algorithm specific to the determination of the greatest state and control.

Algorithm 1

Step 0 (initialization) : $\lambda(k) \leftarrow T$ for $k \geq k_s$; $\mu(k_s) \leftarrow x_{k_s}$ and $\mu(k) \leftarrow T$ for $k \geq k_s + 1$;

Repeat

- Step 1 : for $k = k_f$ to k_s, $\lambda(k) \leftarrow (A_0 \oplus A_0^-)^* \backslash [\mu(k) \wedge [(A_1 \oplus A_1^-) \backslash \lambda(k+1)] \wedge [C \backslash z(k)]]$
- Step 2 : for $k = k_f$ to k_s, $u(k) \leftarrow (B \oplus B^-) \backslash \lambda(k)$
- Step 3 : $\mu(k_s) \leftarrow \lambda(k_s)$

for $k = k_s + 1$ to k_f, repeat $\mu(k) \leftarrow \lambda(k) \wedge [A_1 \otimes \mu(k-1) \oplus A_0 \otimes \mu(k) \oplus B \otimes u(k)] \wedge f^+(\mu(k-1), \mu(k), u(k))$ until no $\mu_i(k)$ changes.

Until $\lambda(k) = \mu(k)$ for $k_s \leq k \leq k_f$ or $\mu(k_s) \neq x_{k_s}$

The introduction of the variables λ and μ allows the memorization of the past values of the state. The "Backward" part corresponds to steps 1 and 2, and the first iteration allows the determination of the starting state trajectory λ of the general algorithm before an optimal minimization. As variables $\mu(k)$ and $\lambda(k)$ are contained in the right-hand terms of the expressions, step 1 results from the application of Theorem 4.73 part 1 in [1] to $\lambda(k) \leftarrow \mu(k) \wedge [(A_1 \oplus A_1^-) \backslash \lambda(k+1) \wedge (A_0 \oplus A_0^-) \backslash \lambda(k)] \wedge [C \backslash z(k)]$ and step 3 improves the state value expressed by μ with a fixed point approach applied to the relevant relation. Finally, when the minimization of the state stops, the algorithm gives the optimal state and control which verifies the inequalities of the model. For instance, the algorithm gives a state trajectory of a Timed Event Graph which verifies the state equation. As the control is directly calculated from the state which is minimized at each step until the convergence, the minimization of the control does not need a memorization of their previous calculated values as for the state.

As the control is applied to a Timed Event Graph whose evolution must start from an initial state, an arbitrary initial condition x_{k_s} is assumed to be known : this "noncanonical" initial condition can be the result of the past evolution of a process, for instance. Therefore, expression $x(k_s) = x_{k_s}$ is a new additional constraint which must be considered in the determination of the control : this constraint is introduced in initialization $\mu(k_s) \leftarrow x_{k_s}$ which is a form of relaxation $x(k_s) \leq x_{k_s}$ because the algorithm calculates the greatest state trajectory and minimizes the initial values. Therefore, the solution presents two cases :

- the estimate of the greatest state trajectory verifies $x^+(k_s) = x_{k_s}$ and therefore, there are a control and state trajectories which satisfy all constraints.

- the estimate of the greatest state trajectory verifies $x(k_s) \neq x_{k_s}$: the previous problem is without solution. If the problem converges to a finite solution with $x(k_s) \leq x_{k_s}$ and $x(k_s) \neq x_{k_s}$, this case can be interpreted as a past evolution of the process whose trajectory is "too early" relatively to x_{k_s}.

The development of the algorithm is easy and requires only the memorization of the matrices of the different models and the estimated trajectory. In the general case, it is difficult to carry out a theoretical analysis of the number of iterations like in many algorithms in this field [6]. The general algorithm of Mc Millan and Dill [26] is known to be pseudo-polynomial [26] [28] in practice. The following result however, shows that the proposed algorithm is very quick when there is no specification : the algorithm converges to the greatest state and control in one iteration. The model of the Timed Event Graph is supposed to be $x(k) = A \otimes x(k-1) \oplus B \otimes u(k)$. In this case, $A_1 \oplus A_1^- = A \oplus A_1^- = A$, $B \oplus B^- = B$ and $f^+(\mu(k-1), \mu(k), u(k)) = +\infty$. The following proposition proves that $\mu(k) \leq [A \backslash \mu(k+1)] \wedge [C \backslash z(k)]$ which entails that any new iteration of the loop as in the general algorithm, will be useless.

Proposition 4.4. *For an optimal control synthesis of a Timed Event Graph without specification, the following inequality is satisfied at the end of step 2 of the algorithm :*

$\mu(k) \leq [(A \oplus A^-) \backslash \mu(k+1)] \wedge [C \backslash z(k)]$

Proof. Simple substitutions and property of distributivity of the left-residual $\backslash$ relative to $\wedge$ are applied below :

$A \backslash \mu(k+1) \wedge C \backslash z(k) = A \backslash [\lambda(k+1) \wedge [A\mu(k) \oplus Bu(k+1)]] \wedge C \backslash z(k)$

$= A \backslash \lambda(k+1) \wedge A \backslash [A\mu(k) \oplus Bu(k+1)] \wedge C \backslash z(k) = A \backslash \lambda(k+1) \wedge C \backslash z(k) \wedge A \backslash [A\mu(k) \oplus Bu(k+1)]$.

So, $A \backslash \mu(k+1) \wedge C \backslash z(k) = A \backslash \lambda(k+1) \wedge C \backslash z(k) \wedge A \backslash [A\mu(k) \oplus Bu(k+1)]$

- Let us consider term $A \backslash \lambda(k+1) \wedge C \backslash z(k)$ in the previous relation. $\lambda(k) = A \backslash \lambda(k+1) \wedge C \backslash z(k)$ at the first iteration because $\mu(k) = T$ on horizon $[k_s+1, k_f]$. However, $\lambda(k_s) = \mu(k_s) \wedge [A \backslash \lambda(k_s+1)] \wedge [C \backslash z(k_s)]$ and consequently, $A \backslash \lambda(k+1) \wedge C \backslash z(k) \geq \lambda(k)$ for k belonging to $[k_s, k_f]$.

Therefore, $A \backslash \mu(k+1) \wedge C \backslash z(k) \geq \lambda(k) \wedge A \backslash [A\mu(k) \oplus Bu(k+1)]$

- Now, consider $A \backslash [A\mu(k) \oplus Bu(k+1)]$. If f is residuated, then it exists h such that $foh \leq Id$ and $hof \geq Id$ (see Preliminaries on residuation) and $A \otimes (A \backslash x) \leq Id$ and $A \backslash [A \otimes x] \geq Id$

Consequently, $A \backslash [A \otimes \mu(k)] \geq \mu(k)$ and $A \backslash [A \otimes \mu(k) \oplus B \otimes u(k+1)] \geq \mu(k)$ by isotony of the left-residual.

- Moreover, by construction, step 3 entails $\lambda(k) \geq \mu(k)$

In short, $A \backslash \mu(k+1) \wedge C \backslash z(k) \geq \lambda(k) \wedge A \backslash [A\mu(k) \oplus Bu(k+1)] \geq \mu(k)$ which terminates the proof.

The previous result shows that the proposed approach generalizes the classical "Backward" approach for Timed Event Graphs. Let us recall that in this case, the greatest solution (the latest times) of the control problem is explicitly given by the “Backward" recursive equations where the co-vector plays the role of the state vector, whereas dater equations give the least solution (the earliest times) of the process evolution [1].

4.8 Example

The following example allows the illustration of the theoretical results. Computation tests are made using Maxplus toolboxes in Scilab. A simple example of Time Stream Petri nets, is given in figure 4.9 and describes the specifications of the Timed Event Graph of figure 4.8.

4.8.1 Models

4.8.1.1 Plant : Timed Event Graph

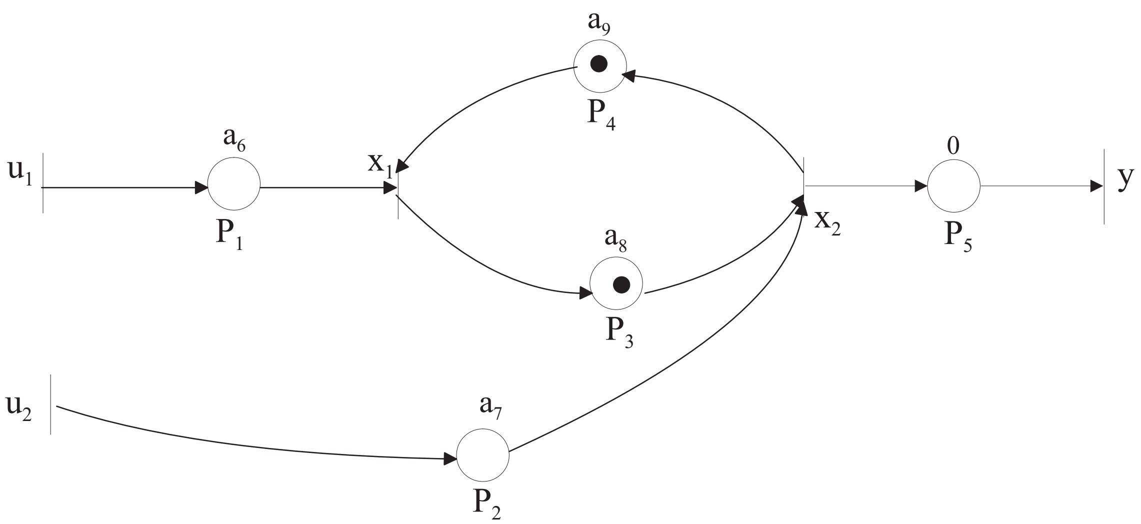

FIG. 4.8 : Timed Event Graph

$x(k) = \begin{pmatrix} x_1(k) \\ x_2(k) \end{pmatrix}$, $u(k) = \begin{pmatrix} u_1(k) \\ u_2(k) \end{pmatrix}$

$x(k) = A_1 \otimes x(k-1) \oplus B \otimes u(k)$ and $y(k) = C \otimes x(k)$ where $A_1 = \begin{pmatrix} \varepsilon & a_9 \\ a_8 & \varepsilon \end{pmatrix}$, $B = \begin{pmatrix} a_6 & \varepsilon \\ \varepsilon & a_7 \end{pmatrix}$ and $C = (\varepsilon \;\; e)$

$a_6 = 6$, $a_7 = 1$, $a_8 = 3$ and $a_9 = 2$

4.8.1.2 Specifications : Time Stream Event Graph

$x(k) \geq f^-(x(k-1), u(k)) = A_1^- \otimes x(k-1) \oplus B^- u(k)$ where

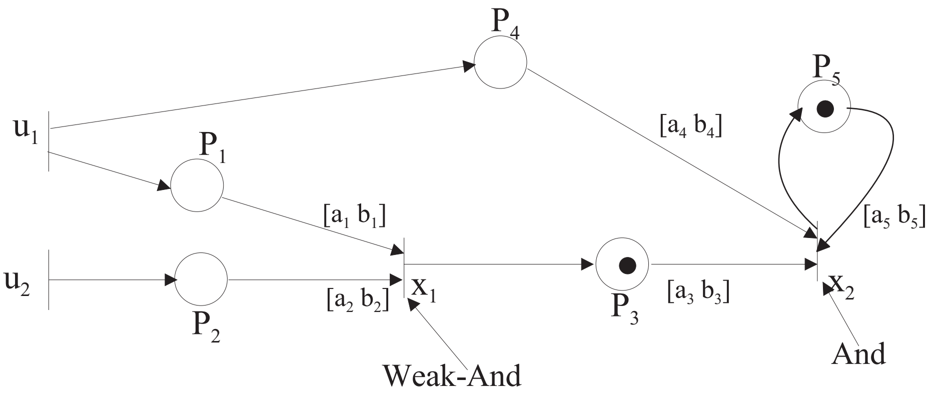

Fig. 4.9 : Time Stream Event Graph

$A_1^- = \begin{pmatrix} \varepsilon & \varepsilon \\ a_3 & a_5 \end{pmatrix}$, $B^- = \begin{pmatrix} a_1 & a_2 \\ a_4 & \varepsilon \end{pmatrix}$.

$x(k) \leq f^+(x(k-1), u(k)) = \bigwedge_{i=1}^{i_{\max}} (A_i^+ \otimes x(k-1) \oplus B_i^+ \otimes u(k))$ where

$i_{\max} = 3$, $A_1^+ = \begin{pmatrix} \varepsilon & \varepsilon \\ b_3 & \varepsilon \end{pmatrix}$, $B_1^+ = \begin{pmatrix} b_1 & b_2 \\ \varepsilon & \varepsilon \end{pmatrix}$, $A_2^+ = \begin{pmatrix} T & T \\ \varepsilon & \varepsilon \end{pmatrix}$, $B_2^+ = \begin{pmatrix} T & T \\ b_4 & \varepsilon \end{pmatrix}$, $A_3^+ = \begin{pmatrix} T & T \\ \varepsilon & b_5 \end{pmatrix}$, $B_3^+ = \begin{pmatrix} T & T \\ \varepsilon & \varepsilon \end{pmatrix}$.

$[a_1\ b_1] = [1, 5]$, $[a_2\ b_2] = [2, 6]$, $[a_3\ b_3] = [2, 11]$, $[a_4\ b_4] = [3, 10]$ and $[a_5\ b_5] = [0, 3]$.

4.8.2 Fixed point formulation

The homogeneous form of (4.11) in Theorem 11 is defined by : $H \begin{pmatrix} x_0 \\ V \end{pmatrix} = H_1 \begin{pmatrix} x_0 \\ V \end{pmatrix} \wedge H_2 \begin{pmatrix} x(k-1) \\ u(k) \end{pmatrix}$

$$H_1 \begin{pmatrix} x_0 \\ x(k-1) \\ x(k) \\ x(k+1) \\ u(k) \end{pmatrix} = \begin{pmatrix} x_0 \\ x(k-1) \\ (A_1 \oplus A_1^-)\backslash x(k+1) \wedge (C\backslash(z(k) \otimes x_0)) \\ x(k+1) \\ (B \oplus B^-)\backslash x(k) \end{pmatrix}$$

$$\begin{pmatrix} x_0 \\ (A_1 \oplus A_1^-)\backslash x(k+1) \wedge (C\backslash(z(k) \otimes x_0)) \\ (B \oplus B^-)\backslash x(k) \end{pmatrix} = \begin{pmatrix} x_0 \\ (a_3 \oplus a_8) \setminus (x_2 k^{-1}) \\ (a_9\backslash(x_1 k^{-1})) \wedge (a_5\backslash(x_2 k^{-1})) \wedge (z \otimes x_0) \\ ((a_6 \oplus a_1) \setminus x_1) \wedge (a_4 \setminus x_2) \\ (a_2 \setminus x_1) \wedge (a_7 \setminus x_2) \end{pmatrix}$$

$$H_2 \begin{pmatrix} x_0 \\ x(k-1) \\ x(k) \\ x(k+1) \\ u(k) \end{pmatrix} = \begin{pmatrix} T \\ T \\ [A_1 \otimes x(k-1) \oplus B \otimes u(k)] \wedge f^+(x(k-1), u(k)) \\ T \\ T \end{pmatrix} \text{ with } A_1 \otimes x(k) \oplus$$

$$B \otimes u(k) = \begin{cases} a_9 \otimes x_2k \oplus a_6 \otimes u_1 \\ a_8 \otimes x_1k \oplus a_7 \otimes u_2 \end{cases}$$

$$f^+(x(k-1), u(k)) = \begin{cases} b_1 \otimes u_1 \oplus b_2 \otimes u_2 \\ (b_3 \otimes x_1k) \wedge (b_4 \otimes u_1) \wedge (b_5 \otimes x_2k) \end{cases}$$

4.8.3 Existence

The only circuit of H_1 is the loop connected to x_0 whose weight equals zero. Let us consider a horizon $l = 6 : X = (x_0, x_1(k_s), x_2(k_s), u_1(k_s), u_2(k_s), ...x_1(k_s+6), x_2(k_s+6), u_1(k_s+6), u_2(k_s+6))^t$. As there is a path from the above circuit to each vertex except $x_1(k_s+6)$, the relevant components in $\chi(\Phi_{H_1}(X_6))$ are equal to zero. The associated graph of H is far more complicated than H_1 and its treatment requires a program specific to dynamic systems. Using Theorem 14, the algorithm of part 4.7 gives a simple approach : as the calculations show the existence of a state and input trajectory for a horizon $[k_s, k_s+6]$, $\chi(\Phi_H(X_6)) \geq 0$. Moreover, $\chi(\Phi_H(X_6)) \leq \chi(\Phi_{H_1}(X_6))$ and there is a path in the associated graph of H, from x_0 to $x_1(k_s+6)$ $(x_1(k) \leq a_9 \otimes x_2(k-1) \oplus a_6 \otimes u_1$ and $x_2(k-1) \leq z(k-1) \otimes x_0)$. Consequently, $\chi(\Phi_H(X_6)) = 0$. Below, some examples of simple circuits illustrate the possible incoherences in the plant (circuit 1) or, between the plant and its specifications (circuits 2 and 3). When a strictly negative circuit appears, the algorithm of control can diverge as indicated in the proof of proposition 4.3 : if condition $\chi(\Phi_H(X_l)) \geq 0$ is not satisfied, Theorem 14 says that there is no solution.

Circuit 1

$$\begin{cases} x_2(k) \leq a_5 \setminus x_2(k+1)) \\ x_2(k+1) \leq (b_5 \otimes x_2(k-1)) \end{cases}$$

For instance, an elementary error in the modeling is the consideration of $a_5 > b_5$ with $a_5 = 10$ and $b_5 = 3$. The weight of the circuit is : $b_5 - a_5$. Therefore, the mean weight of the circuit is negative and equals to $-7/2$. The calculation of the spectral vector with an algorithm developed with Scilab, confirms this result and shows the incoherence of the process on horizon l=1 : $\chi(\Phi_H(X_1)) = (0, -3.5, -3.5, -3.5, -3.5, -3.5, -3.5, -3.5, -3.5)^t$

Circuit 2

$$\begin{cases} x_1(k) \leq ((a_3 \oplus a_8) \setminus x_2(k+1)) \\ x_2(k+1) \leq (b_3 \otimes x_1(k)) \end{cases}$$

The weight of the circuit is : $b_3 - (a_3 \oplus a_8) = 11 - (2 \oplus 3) = 8$

Now, if $a_8 = 12$, the mean weight of circuit is negative and equals to -0.5. The calculation of the spectral vector confirms this result and shows that the process cannot follow the specifications on horizon l=1 : $\chi(\Phi_H(X_1)) = (0, -0.5, -0.5, -0.5, -0.5, -0.5, -0.5, -0.5, -0.5)^t$

Circuit 3

$$\begin{cases} x_2(k) \leq b_4 \otimes u_1(k) \\ u_1(k) \leq (a_6 \oplus a_1) \setminus x_1(k)) \\ x_1(k) \leq (a_3 \oplus a_8) \setminus x_2(k+1) \\ x_2(k+1) \leq b_5 \otimes x_2(k) \end{cases}$$

The weight of the circuit is : $b_4 - (a_6 \oplus a_1) - (a_3 \oplus a_8) + b_5 = 10 - (6) - 3 + 3 = 13 - 9 = 4$

Now, if $b_4 = 5$, the mean weight of circuit is negative and equals to -0.25. The calculation of the spectral vector confirms this result and shows that the process cannot follow the specifications on horizon l=2 :

$\chi(\Phi_H(X_2)) = (0, -0.25, -0.25, -0.25, -0.25, -0.25, -0.25, -0.25, -0.25, -0.25, -0.25, -0.25, -0.25)^t$

4.8.4 Optimal control with specifications

The desired output z is as follows.

k	k_s	k_s+1	k_s+2	k_s+3	k_s+4	k_s+5	k_s+6
z	30	31	32	33	34	35	35

The resolution of part H_1 gives :

k	k_s	k_s+1	k_s+2	k_s+3	k_s+4	k_s+5	k_s+6
x_1	22	22	27	27	32	32	T
x_2	20	25	25	30	30	35	35

k	k_s	k_s+1	k_s+2	k_s+3	k_s+4	k_s+5	k_s+6
u_1	16	16	21	21	26	26	32
u_2	19	20	24	25	29	30	34

Now, the complete model H is considered. The following table gives the greatest trajectories of the state and the control for different initial conditions. Let us note that the previous greatest solution gives a finite upper bound of the following solutions which satisfied $V = h(V)$ (see proposition 4.2). Equally, the algorithm verifies that the following trajectories verifies the model of the Timed Event Graph and its specifications.

For $x_{k_s} = \begin{pmatrix} T \\ T \end{pmatrix}$,

k	k_s	k_s+1	k_s+2	k_s+3	k_s+4	k_s+5	k_s+6
x_1	20	22	25	27	30	32	38
x_2	20	23	25	28	30	33	35

k	k_s	k_s+1	k_s+2	k_s+3	k_s+4	k_s+5	k_s+6
u_1	14	16	19	21	24	26	32
u_2	18	20	23	25	28	30	34

The above trajectories represent the greatest trajectories which cannot be exceeded.

4.8.5 Initial conditions

Now, the consistency or admissibility, of $x(k_s) = x_{k_s}$ is considered. For $x_{k_s} = \begin{pmatrix} 19 \\ 20 \end{pmatrix}$,

k	k_s	k_s+1	k_s+2	k_s+3	k_s+4	k_s+5	k_s+6
x_1	19	22	25	27	30	32	38
x_2	20	22	25	28	30	33	35

Therefore, $x(k_s) = x_{k_s}$ is admissible.

k	k_s	k_s+1	k_s+2	k_s+3	k_s+4	k_s+5	k_s+6
u_1	13	16	19	21	24	26	32
u_2	17	20	23	25	28	30	34

For $x_{k_s} = \begin{pmatrix} 18 \\ 20 \end{pmatrix}$,

k	k_s	k_s+1	k_s+2	k_s+3	k_s+4	k_s+5	k_s+6
x_1	18	21	24	27	30	32	38
x_2	19	21	24	27	30	33	35

Therefore, $x(k_s) \neq x_{k_s}$is not admissible.

k	k_s	k_s+1	k_s+2	k_s+3	k_s+4	k_s+5	k_s+6
u_1	12	15	18	21	24	26	32
u_2	16	19	22	25	28	30	34

4.9 Conclusion

In this chapter, we have shown that different Time Event Graphs can be modeled in the form of a new model called time interval system which is based on (min, max, $+$) functions. The interval descriptor system can describe the time behavior of a lot of models such as Timed Event Graphs, P-time Petri nets and Time Stream Event Graphs. These results lead us to think that the interval descriptor system will bring a unified model and a new subject in the field of Time and Timed Petri net.

The following part solves the problem of optimal control synthesis of a Timed Event Graph when the state and control trajectories are constrained by specifications defined by an interval model. Considering the semantic rules “And” and “Weak-And” in the case of a Time Stream

Event Graphs, the problem is reformulated under a fixed point form. The spectral theory gives conditions of existence of a solution which verifies the corresponding equality when the Event Graph is structurally observable. A proposed algorithm which is pseudo polynomial makes it possible to determine the greatest state and control when the Timed Event Graph starts from an arbitrary initial condition. The application of the calculated control generates a state trajectory obeying the specifications.

References

[1] S. Amari, I. Demongodin, and J.J. Loiseau, 2005. Control of Linear min-plus Systems under Temporal Constraints. *Proc. 44th IEEE Conference on Decision and Control and European Control Conference ECC, CDC-ECC'05.* Seville, Spain, pp.7738-7743.

[2] P. Aygalinc, S. Calvez, W. Khansa and S. Collart-Dutilleul, *Using P-time Petri Nets for Robust Control of Manufacturing Systems*, 1st IFAC-Workshop on Manufacturing Systems (MIM'97), Wien, Austria, February 1997, pp. 75-80.

[3] F. Baccelli, G. Cohen, G.J. Olsder J.P. and Quadrat, *Synchronization and Linearity. An Algebra for Discrete Event Systems*, available from http ://maxplus.org, New York, Wiley, 1992.

[4] G. Cohen, S. Gaubert, and J.-P. Quadrat, *From first to second-order theory of linear discrete event systems*, 1st SIAC World Congress, Sydney, Australia, 1993.

[5] Y. Cheng and D.-Z. Zheng, *On the cycle Time of Non-autonomous Min-max Systems*, IEEE, 2002.

[6] Y. Cheng and D-Z. Zheng, *Min-max inequalities and the timing verification problem with max and linear constraints*, Discrete Event Dynamic Systems : Theory and Applications, Springer Science and Business Media, Vol 15, pp. 119–143, 2005.

[7] B. Heidergott, G. J. Olsder and L. van der Woude, *Max Plus at Work*, Princeton University Press, 2006.

[8] J. Cochet-Terrasson, S. Gaubert and J. Gunawardena, *A constructive fixed point theorem for min-max functions, Dynamics and Stability of Systems*, Vol. 14 N°4, 1999 pp. 407-433.

[9] J. Cochet-Terrasson *Algorithmes d'itération sur les politiques pour les applications monotones contractantes*, thèse, Ecole Nationale Supérieure des Mines de Paris, 2001.

[10] S. Collart-Dutilleul and P. Yim, Time window specification and validation with Petri nets, ETFA, IEEE Conference on Emerging Technologies and Factory Automation, September 16-19, 2003, Lisbon, Portugal, pp. 232-237.

[11] J-P. Courtiat, M. Diaz, R.C. De Oliviera and P. Sénac *Formal models for the description of timed behaviors of multimedia and hyper media distributed systems*, Computer Communications 19, 1996, pp. 1134-1150.

[12] B.A. Davey and H.A. Priestley, *Introduction to lattices and order*, Cambridge University Press, Cambridge, UK, 2002.

[13] P. Declerck and M.K. Didi Alaoui, *Optimal Control Synthesis in Interval Descriptor Systems, Application to Time Stream Event Graphs*, IFAC Congress Praha, 2005.

[14] P. Declerck and M.K Didi Alaoui, *Modeling and liveness analysis of P-time event graphs in the (min, max, +) algebra*, IEEE International Conference on Systems, Man and Cybernetics, The Hague, The Netherlands, October 10-13, 2004.

[15] Declerck, P., and Didi Alaoui, M.K. 2005. Extremal trajectories in P-time Event Graphs : application to control synthesis with specifications. *Proc. 44th IEEE Conference on Decision and Control and European Control Conference ECC, CDC-ECC'05*. Seville, Spain, pp.7621-7626.

[16] M. Diaz and P. Owezarski *From multimedia models to multimedia transport protocols*, Computer Networks and ISDN Systems 29, 1997, pp. 745-758.

[17] M. Diaz *Les réseaux de Petri : Modèles fondamentaux*, Hermès, Paris, France, 2001.

[18] M.K. Didi Alaoui, P. Declerck *From (min, max, +) algebra to P-time event graphs*, ICINCO'04 International Conference on Informatics in Control, Automation and Robotics Setubal Portugal 2004.

[19] Didi Alaoui, M.K. 2005. Etude et supervision des graphes d'événements temporisés et temporels : vivacité, estimation et commande. *PhD Thesis*, available from http ://www.istia.univ-angers.fr/

[20] S. Gaubert and J. Gunawardena, *the duality theorem for min-max functions* , CRAS, t. 326, Série I, 1998, pp. 43-48.

[21] M. Gondran and M. Minoux *Graphes, dioïdes et semi-anneaux*, Tec & Doc, Paris, France, 2001.

[22] A. Guezzi, Ph. Declerck et J.-L. Boimond, Commande de graphes d'événements temporisés sur un horizon glissant, Journal Européen des Systèmes Automatisés (JESA), vol. 43/7-9, p. 1097-1111, 2009.

[23] A. Guezzi, Modélisation, analyse de performance et commande des systèmes à événements discrets, Thèse de doctorat, Université d'Angers, disponible sur http ://www.istia.univ-angers.fr/, 2010.

[24] S. Hashtrudi Zad, R.H. Kwong and W.M. Wonham, *Supremum operators and computation of supremal elements in system theory*, Siam J. Control Optm., Vol.37, No. 3, 1999, pp. 695-709.

[25] R. Kumar and V.K Garg, *Extremal solutions of inequations over lattices with applications to supervisory control*, Theoret. Comp. Sci., 148, 1995, pp. 67-92.

[26] K. Mc Millan and D. Dill *Algorithms for interface timing verification*, Proceedings of the IEEE, International Conference on Computer Design : VLSI in Computers and Processors, 1992.

[27] B. De Schutter and T. van den Boom, *Model predictive control for max-plus-linear discrete event systems*, Automatica, Vol. 37(7), 2001 pp. 1049-1056.

[28] E. Walkup, *Optimization of linear max-plus systems with application to timing analysis*, University of Washington, PhD thesis, 1995.

Chapitre 5

Commande prédictive sur un horizon glissant

L'objectif de cette partie est la commande prédictive d'un Graphe d'Evénements Temporisé avec pour spécifications, un ou des Graphes d'Evénements P-temporels. Nous restons ici dans l'algèbre (max, +). Contrairement au chapitre précédent sur la commande, nous mettons l'accent sur l'obtention de la commande sur un horizon glissant. Ce mode en ligne impose l'utilisation d'une procédure efficace sous peine d'obtenir un champ réduit d'application en termes de dimensions du systèmes et d'horizons. Exploitant au maximum la structure spécifique des systèmes, le temps CPU de la procédure en-ligne est réduit fortement par deux techniques : l'utilisation d'algorithmes spécifiques de la théorie des graphes au lieu d'algorithmes génériques ; une préparation hors-ligne qui évite la répétition de calculs identiques.

5.1 Introduction

5.1.1 Problem

The aim of this chapter is to develop a model predictive control on a sliding horizon. The specifications are defined by P-time Event Graphs which describe the desired behavior of the interconnections of all the internal transitions. P-time Event Graphs concern time-constrained systems where the duration of each operation is bounded by minimal and maximal limits. Different to the usual rule applied to Timed Event Graphs, the firing of the transitions is not "as soon as possible" otherwise some tokens can die which can lead to a deadlock situation.

Naturally, this topic has already been considered in different studies in the Netherlands ([36]) where a usual step is to transform the (max, +) problem into a linear programming problem in the conventional algebra which allows the application of classical algorithms. The principal advantage of this technique is the consideration of different classes of models. However, model predictive control is an on-line approach which needs *efficient algorithms* : indeed, a crucial point is that a too slow calculation of the control can postpone the application of the control at the calculated dates.

The ELCP algorithm (Extended Linear Complementary Problem) described in chapter 3 of the thesis of Bart de Schutter cannot be used for on-line computations as the CPU time increases exponentially ([36]). The algorithms of Khashiyan and Karmarkar in linear programming are famous but it is known that they are polynomial *in the weak sense* ([35]) and not in the strong sense (contrary to many algorithms in graph theory : see chapter 2 in ([24])). The complexity of the Simplex is exponential in the worst case even if this algorithm is relatively good on the average ([33]). As a consequence, the application of these *generic* algorithms of linear programming (see the algorithms quoted in ([36])) leads to the limitation of the size of the considered systems and the magnitude of the entries. Another difficulty is that these generic algorithms correspond to the optimisation stage which needs to start from an admissible solution which is the result of another stage. Not immediate, this determination can be included into the program (function simplex "linpro()" of Scilab) or not (function "karmarkar(a,b,c,x0)" of Scilab where x0 is the initial vector). Therefore, a problem is the determination of an admissible solution close to the optimal solution.

5.1.2 General answer

In fact, the crucial point is that the structure of the matrices considered above present *specific characteristics* : the matrices are sparse and contain many rows with two non-null entries (1 and -1) at the most. The matrices are close to the ingoing/outgoing incidence matrices of the fundamental marking relation ([26]). Therefore, the goal of the chapter is to make the most of these specific structures of the systems and to deduce an approach having a reduced CPU time.

A general answer is to use the (max, +) algebra which allows the application of efficient algorithms of path theory : in general, they are *strongly* polynomial (the running time does not depend on the magnitudes of the parameters). A direct consequence is that these algorithms specific to path algebra, surpass the best generic algorithms of linear programming when they are applied to the specific problems : these algorithms can consider large scale systems. Our tests show that, if the simplex is used to calculate the maximal paths between every pair of vertices (function "linpro()"), the approach can work until 60 vertices and needs about 2 minutes while the (max, +) kleene star (function "star()") solves the problem in 1 second for 300 vertices. Moreover, the (max, +) algebra allows the writing of elegant expressions contrary to linear programming which can only give the numerical results.

Another improvement is as follows. In fact, the predictive control is made on a sliding horizon : the horizon is slightly moved back at each step and the control is calculated. The idea is to avoid the repetition of the same calculations at each step which can be costly in terms of time. Before the application of the on-line control, a *preparation* can contain these calculations allowing a reduction of the complexity of the on-line procedure.

The chapter is structured as follows : In the first part, we consider the control problem without desired output and with a fixed horizon and the determination of the earliest trajectory (problem 1). In the following part, the desired output defined on a fixed horizon is introduced and a just-in-time control is made. The problem (problem 2) is to determine the greatest input in order to obtain a desired behavior defined by the desired output and the specifications. In the following part, the approach is generalized to a sliding horizon (problem 3) and we consider the CPU time of the on-line procedure based on a preparation using the Kleene star of a large scale tri-diagonal matrix. Finally, the structure analysis of this special leads to a more efficient technique of calculation of the Kleene star.

Recall that maximization, minimization and addition operations are denoted respectively $\oplus$, $\wedge$ and $\otimes$. A matrix is called row-astic if it has no null row.

5.2 Control without desired output (problem 1)

5.2.1 Objective

The problem of this part is the control of a plant described by a Timed Event Graph when the state and control trajectories are constrained by additional specifications defined by a P-time Event graph. The objective is the determination of an admissible (arbitrary) control u on horizon $[k_s+1, k_f]$ such that its application to the Timed Event Graph defined by

$$\begin{cases} x(k+1) = A \otimes x(k) \oplus B \otimes u(k+1) \\ y(k) = C \otimes x(k) \end{cases} \tag{5.1}$$

satisfies the following conditions :

a) The state trajectory follows the model of the autonomous P-time Event Graph defined by

$$\begin{pmatrix} x(k) \\ x(k+1) \end{pmatrix} \geq \begin{pmatrix} A^{=} & A^{+} \\ A^{-} & A^{=} \end{pmatrix} \otimes \begin{pmatrix} x(k) \\ x(k+1) \end{pmatrix} \tag{5.2}$$

for $k \geq k_s$; matrix A^- (respectively, A^+) contains the lower bounds of the temporizations (respectively, the upper bounds with minus sign) associated with each place having a unitary initial marking. We have $A^{=} = A_0^- \oplus A_0^+$ where matrix A_0^- (respectively, A_0^+) is defined as A^- (respectively, A^+) but with with a null initial marking. See chapter ”Vivacité” for more details.

b) The first state vector of the state trajectory $x(k)$ for $k \geq k_s$ is finite and is a known vector denoted $\underline{x}(k_s)$. This “ non-canonical ” initial condition can be the result of a past evolution of a process. As $x(k_s)$ is finite, the trajectories considered in this chapter are finite.

Underlined symbols like $\underline{x}(k_s)$ correspond to known data of the problem and $x(k)$ and $y(k)$ are estimated in the following resolutions.

Therefore, we focus on a control problem without desired output on a fixed horizon. This problem is denoted Problem 1.

The following example shows that the space solution to Problem 1 is not an inf-semilattice that is, Problem 1 has no (unique) minimal solution by respect to the componentwise order.

5.2.2 Example 1

Timed Event Graph :
$A = \begin{pmatrix} T_1 & \varepsilon \\ T_2 & T_3 \end{pmatrix}$, $B = \begin{pmatrix} T_4 & T_5 & \varepsilon \\ \varepsilon & T_6 & T_7 \end{pmatrix}$ and $C = \begin{pmatrix} \varepsilon & T_8 \end{pmatrix}$.
P-Time Event Graph :
$A^{=} = \varepsilon$, $A^{-} = \begin{pmatrix} T_{10} & T_{13} \\ T_{11} & T_{12} \end{pmatrix}$ and $A^{+} = \begin{pmatrix} -T_{20} & -T_{21} \\ -T_{23} & -T_{22} \end{pmatrix}$. Every temporisation is finite except $T_{20} = T_{21} = T_{22} = T_{23} = +\infty$.

For $k_f = k_s + 1$, we have $x(k_s+1) = A \otimes \underline{x}(k_s) \oplus B \otimes u(k+1) \geq A^{-} \otimes \underline{x}(k_s)$. For $\underline{x}(k_s) = \begin{pmatrix} 0 & 0 \end{pmatrix}^t$, $\begin{cases} x_1(k_s+1) = T_1 \oplus T_4 \otimes u_1(k_s+1) \oplus T_5 \otimes u_2(k_s+1) \\ \geq T_{10} \oplus T_{13} \\ x_2(k_s+1) = T_2 \oplus T_3 \oplus T_6 \otimes u_2(k_s+1) \oplus T_7 \otimes u_3(k_s+1) \\ \geq T_{11} \oplus T_{12} \end{cases}$

If $T_1 < T_{10} \oplus T_{13}$ and $T_2 \oplus T_3 < T_{11} \oplus T_{12}$, the system becomes :
$\begin{cases} T_4 \otimes u_1(k_s+1) \oplus T_5 \otimes u_2(k_s+1) \geq T_{10} \oplus T_{13} \\ T_6 \otimes u_2(k_s+1) \oplus T_7 \otimes u_3(k_s+1) \geq T_{11} \oplus T_{12} \end{cases}$. This system has no (unique) minimal solution by respect to the componentwise order : if $u_2(k_s+1) = \varepsilon$, $u_1(k_s+1)$ and $u_3(k_s+1)$ must be finite ; if $u_1(k_s+1) = \varepsilon$ and $u_3(k_s+1) = \varepsilon$, $u_2(k_s+1)$ must be finite. ■

In the following part 5.2.3, we present the relations which describe a trajectory of a Timed Event Graph satisfying the specifications defined by a P-time Event Graph (constraint a)).

5.2.3 Trajectory description

From (5.1) and (5.2), we deduce a system which describes the trajectories on horizon $[k_s, k_f]$. Let us introduce the following notations. Let $X = \begin{pmatrix} x(k_s)^t & x(k_s+1)^t & x(k_s+2)^t & \cdots & x(k_f-1)^t & x(k_f)^t \end{pmatrix}^t$ (t : transposed) and $D_h =$

$$\begin{pmatrix} A^{=} & A^{+} & \varepsilon & \cdots & \varepsilon & \varepsilon & \varepsilon \\ A \oplus A^{-} & A^{=} & A^{+} & \cdots & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & A \oplus A^{-} & A^{=} & \cdots & \varepsilon & \varepsilon & \varepsilon \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \varepsilon & \varepsilon & \varepsilon & \cdots & A^{=} & A^{+} & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \cdots & A \oplus A^{-} & A^{=} & A^{+} \\ \varepsilon & \varepsilon & \varepsilon & \cdots & \varepsilon & A \oplus A^{-} & A^{=} \end{pmatrix}$$ with

$h = k_f - k_s$. Matrix D_h presents an original block tri-diagonal structure : this is a square matrix, composed of a lower diagonal (square sub-matrices $A \oplus A^{-}$), a main diagonal (square sub-matrices $A^{=}$) and an upper diagonal (square sub-matrices A^{+}), with all other blocks being zero matrices (ε). Matrix D_h is a $n.(h+1)$) x $n.(h+1)$) matrix where n is the dimension of x.

Theorem 15. *(Theorem 2 in ([17])) The state trajectories of a Timed Event Graph (5.1) starting from* $\underline{x}(k_s)$ *and following the specifications defined by a P-time Event Graph (5.2) on horizon* $[k_s, k_f]$ *satisfy the following system*

$$\begin{cases} X \geq D_h \otimes X \\ x(k) \geq B \otimes u(k) \text{ for } k \in [k_s + 1, k_f] \\ x(k) \leq A \otimes x(k-1) \oplus B \otimes u(k) \text{ for } k \in [k_s + 1, k_f] \\ x(k_s) = \underline{x}(k_s) \end{cases} \tag{5.3}$$

Remark 16. *It is important to note that system (5.3) cannot be rewritten in a fixed point form which can be analyzed by known results.*

5.2.4 Relaxed system

Equality $x(k) = A \otimes x(k-1) \oplus B \otimes u(k)$ comes from the earliest firing rule. In this part, we determine the conditions such that the determination of the trajectory only needs to use $x(k) \geq A \otimes x(k-1) \oplus B \otimes u(k)$. Therefore, relation $x(k) \leq A \otimes x(k-1) \oplus B \otimes u(k)$ in (5.3) is disregarded. From system (5.3) which describes Problem 1, we deduce the following relaxed system

$$\begin{cases} X \geq D_h \otimes X \\ x(k_s) = \underline{x}(k_s) \end{cases} \tag{5.4}$$

which presents a fixed point form.

We now characterize the sets of trajectories of systems (5.3) and (5.4).

Property 6. *Each trajectory of system (5.3) satisfies (5.4).*

Proof. Immediate : As system (5.3) contains an additional constraint, any trajectory of this system satisfies relaxed system (5.4). ■

The extended space defined by (5.4) is now restricted by an additional condition $B \otimes u(k) = x(k)$. Contrary to the initial space (see example 1), the space is now an inf-semilattice allowing the use of the efficient algorithms of graph theory. The following theorem is the starting point of the proposed approach. As we below consider the equality $B \otimes u(k) = x(k)$ with $x(k)$ finite for $k \in [k_s + 1, k_f]$, matrix B is necessarily row-astic.

Theorem 16. *(Theorem 4 in ([17])) A trajectory* X *of (5.4) satisfies (5.3) if control* $u(k)$ *satisfies condition* $B \otimes u(k) = x(k)$ *for* $k \in [k_s + 1, k_f]$.

Therefore, an (arbitrary) admissible trajectory satisfying Problem 1 can be found if we can find a trajectory satisfying the relaxed system (5.4) under the condition of existence of a control such that $B \otimes u(k) = x(k)$ for $k \in [k_s + 1, k_f]$. We can also focus on the earliest state trajectory.

Let us now determine the earliest state trajectory X^- with (5.4). Let $E = \begin{pmatrix} \underline{x}(k_s)^t & \varepsilon & \cdots & \varepsilon \end{pmatrix}^t$ with $dim(E) = dim(X)$. As constraint $x(k_s) = \underline{x}(k_s)$ can be written $x(k_s) \leq \underline{x}(k_s)$ and $\underline{x}(k_s) \leq x(k_s)$, the earliest state trajectory X^- is given by the resolution of $X \geq D_h \otimes X \oplus E$ with condition $\underline{x}(k_s) \geq x^-(k_s)$. The application of Kleene star by Theorem 4.75 part 1 in ([1]) gives the lowest solution $X^- = (D_h)^* \otimes E$ with condition $\underline{x}(k_s) \geq x^-(k_s)$. Moreover, a control can be easily calculated : the greatest control to $B \otimes u(k) = x(k)$ is obviously $u(k) = B \backslash x(k)$ under the condition $B \otimes (B \backslash x(k)) = x(k)$.

Let us now make a brief analysis of $B \otimes u(k) = x(k)$. The analysis of this condition is out the scope of this chapter and a deeper study will be proposed in a future chapter. A first analysis on the condition $B \otimes u(k) = x(k)$ is given in ([17]). In the following property, the considered set of matrices includes the set of permutation matrices without being limited to this set.

Property 7. *If each row i of the matrix B at least contains a non-null element $B_{i,j}$ which is unique in the column j , we can find a control $u(k)$ such that $B \otimes u(k) = x(k)$ is satisfied for any $x(k)$.*

Proof. If B is row-astic, each row i contains a non-null entry B_{ij} at least and a row can contain more than one element. Suppose that there is an element $B_{i,j}$ which is unique in the column j . As a result of residuation is $(A \backslash b)_i = j = 1 \bigwedge^{m} A_{ji} \backslash b_j$ where A is an $m \times n$ matrix, we obtain $u_j = (B \backslash x(k))_j = B_{ij} \backslash x_i(k)$ for row i and equality $B_{ij} \otimes u_j = x_i(k)$ is satisfied for any state. As the other entries $B_{ij'}$ of the row i satisfy $B_{ij'} \otimes u_{j'} \leq x_i(k)$, there is a control such that $B_{i,.} \otimes u(k) = x_i(k)$ for any state. The generalisation to all rows of B is immediate. Note that different values of control vector can also satisfy condition $B \otimes u(k) = x(k)$. ■

5.3 Control with desired output (problem 2)

5.3.1 Objective

In Problem 1, the objective is to calculate a state and a control trajectory of a Timed Event Graph (5.1) starting from $\underline{x}(k_s)$ and following specifications defined by a P-time Event Graph (5.2) on horizon $[k_s, k_f]$. In this part, we consider the "Just-in-time" objective and we focus on the greatest state and control trajectory where the following condition must be satisfied : $y \leq \underline{z}$ (denoted condition c)) knowing the trajectory of the desired output $\underline{z}$ on a fixed horizon $[k_s + 1, k_f]$ with $h = k_f - k_s \in \mathbb{N}$. This problem is denoted Problem 2.

5.3.2 Fixed point form

Using the previous description of the state and control trajectories (5.3), we rewrite the problem under a general fixed point formulation $x \leq f(x)$ which allows the resolution of Problem 2. Let $X^+ =$

$\begin{pmatrix} x^+(k_s)^t & x^+(k_s+1)^t & x^+(k_s+2)^t & \cdots & x^+(k_f-1)^t & x^+(k_f)^t \end{pmatrix}^t$ be the greatest estimate of state trajectory X.

Theorem 17. *(Theorem 3 in ([17])) The greatest state and control trajectory of Problem 2 is the greatest solution of the following fixed point inequality system*

$$\begin{cases} X \leq D_h \backslash X \\ u(k) \leq B \backslash x(k) \text{ for } k \in [k_s+1, k_f] \\ x(k) \leq [A \otimes x(k-1) \oplus B \otimes u(k)] \wedge C \backslash \underline{z}(k) \\ \text{ for } k \in [k_s+1, k_f] \\ x(k_s) \leq \underline{x}(k_s) \end{cases} \tag{5.5}$$

with condition $\underline{x}(k_s) \leq x^+(k_s)$.

If condition $\underline{x}(k_s) \leq x^+(k_s)$ is satisfied, then $\underline{x}(k_s) = x^+(k_s)$. Note that inequality (5.5) is equivalent to inequality (5.3) if we add the constraint $C \otimes x(k) \leq \underline{z}$. Therefore, the calculated state trajectory for $k \geq k_s$ is consistent with the past evolution $k \leq k_s$: In other words, the Timed Event Graph can follow calculated trajectory X^+ after k_s which obeys the specifications defined by the P-time Event Graph.

System (5.5) leads to a fixed-point formulation whose general form is such that $x \leq f(x)$ where f is a (min, max, +) function and can be defined by the following grammar : $f = b, x_1, x_2, \ldots, x_n \mid f \otimes a \mid f \wedge f \mid f \oplus f$ where a, b are arbitrary real numbers ($a, b \in \mathbb{R}$). It is important to note that the concept of extremal solution exists in system (5.5) contrary to system (5.3).

Effective calculation of the greatest control can be made by a classical iterative algorithm of ([32]) (pseudo-polynomial) which particularizes the algorithm of Kleene to (min, max, +) expressions, or more complex algorithms ([39], [7]).

5.3.3 Relaxed system

The following result shows that assumption $x(k) = B \otimes u(k)$ gives a simplified form $x \leq f(x)$ where f is only a (min,+) function.

Theorem 18. *The greatest state trajectory X^+ of Problem 2 is the greatest solution of the following fixed point inequality system*

$$\begin{cases} X \leq D_h \backslash X \\ x(k) \leq C \backslash \underline{z}(k) \text{ for } k \in [k_s+1, k_f] \\ x(k_s) \leq \underline{x}(k_s) \end{cases} \tag{5.6}$$

with condition $\underline{x}(k_s) \leq x^+(k_s)$ if control $u(k)$ satisfies condition $B \otimes u(k) = x(k)$ for $k \in [k_s+1, k_f]$.

Proof

Theorem 16 shows that a trajectory X of (5.4) satisfies (5.3) if control $u(k)$ satisfies condition $B \otimes u(k) = x(k)$ for $k \in [k_s + 1, k_f]$. This result holds if we add the constraint $C \otimes x(k) \leq \underline{z}$. Therefore, we can deduce system (5.6) like in the previous theorem. ■

Let us calculate the greatest state of the control problem. Let
$F = \begin{pmatrix} \underline{x}(k_s)^t & (C \backslash \underline{z}(k_s+1))^t & (C \backslash \underline{z}(k_s+2))^t & \cdots \end{pmatrix}$
$(C \backslash \underline{z}(k_f))^t)^t$. System of inequalities (5.6) becomes

$$\begin{cases} X \leq D_h \backslash X \wedge F \\ x(k) = u(k) \text{ for } k \in [k_s+1, k_f] \end{cases}$$

with condition $\underline{x}(k_s) \leq x^+(k_s)$. The application of Theorem 4.73 in ([1]) gives the greatest solution $X^+ = D_h^* \backslash F$ with condition $\underline{x}(k_s) \leq x^+(k_s)$.

Similarly to the calculation of the earliest state trajectory, the resolution of relaxed fixed-point form (5.6) uses the Kleene star which improves the resolution : now, the complexity is not pseudo-polynomial but polynomial in the strong sense.

Up to now, Problem 1 and 2 are considered on a fixed horizon. The aim of the following part is the extension of problem 2 to a control approach on a sliding horizon.

5.4 Control on a sliding horizon (problem 3) : on-line and off-line aspects

Let us briefly recall the technique of the predictive control on a sliding horizon. We assume that each event date of transition firing is available for current number of event k : at step $k = k_s$, $\underline{u}(k_s)$ and $\underline{x}(k_s)$ are known. A future control sequence $u(k)$ for $k \in [k_s+1, k_s+h]$ is calculated and only the first element of the optimal sequence (here $u(k_s+1)$) is applied to the process. At the next number of event k_s+1, the horizon is shifted : at step k_s+1, the problem is updated with new information $\underline{u}(k_s+1)$ and $\underline{x}(k_s+1)$ and a new calculation is performed.

However, we must guarantee the coherence of the state trajectory between each iteration, that is, the future trajectory $k \geq k_s+1$ is the extension of the past trajectory $(k \leq k_s)$: more formally, the equality $\underline{x}(k_s) = x^+(k_s)$ must be satisfied otherwise the control problem 2 has no solution. If we consider that the models of the Timed Event Graph and the specifications cannot be modified, a possibility is to put the desired output back such that this problem 2 presents a solution. In this chapter, we choose to solve the problem 2 with the modified desired output trajectory $z^m(k) = \underline{z}(k) \oplus y^-(k)$ for $k \in [k_s+1, k_s+h]$ where y^- is an admissible trajectory. This procedure yields an optimal control for z^m which can always be applied to the process. Indeed, if $u^-$and X^- are the control and the state corresponding to the admissible trajectory y^-, then the control and the state corresponding to z^m are greater.

5.4.1 CPU time of the on-line control

As control approaches using a sliding horizon need efficient algorithms, we consider the complexity of the calculation of the on-line control. We made computation tests on CPU time of the proposed approach using the max-plus toolbox in Scilab 3.1.1 with an Intel Core2 Duo 2.26 GHz. This time value includes the time of the two algorithms corresponding to the prediction procedure and the control synthesis proposed in this chapter which correspond respectively to the resolution of the relaxed systems (5.4) and (5.6) of problems 1 and 2. However, the CPU time of the control does not include the calculation of the star $(D_h)^*$ which is made in the *off-line preparation* which depends on the matrices of the model and the size of the horizon. It needs the memorization of a large matrix $((h+1).n$ x $(h+1).n)$. Therefore, the calculations of the on-line control are only limited to the multiplication of matrices $(D_h)^*$ and E $(X^- = (D_h)^* \otimes E)$ and the left $\otimes-$residuation of F by D_h^* $(X^+ = D_h^* \backslash F)$ in time $O(q^2)$ where $q = (h+1).n$. In the curve of Figure 5.1, we have listed the CPU time needed to compute the control for horizon $h = 50$ and different dimensions n of the state matrices of the event graph until $n = 97$. The matrices of the system are randomly generated ($A^= = \varepsilon$, A^- and A^+ are full). Note that for $n = 90$, the off-line preparation which is the calculation of the kleene star with function star() of Scilab, approximately needs 2.10^3 seconds while the on-line procedure only needs 0.28 seconds (Figure 5.1). Therefore, the CPU time of the on-line control is drastically reduced.

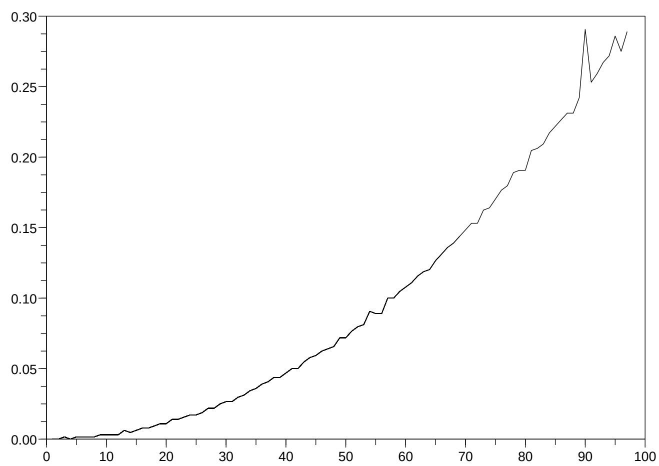

FIG. 5.1 : On-line control : CPU times of one step for h=50 and randomly generated matrices until $n = 97$ transitions. The CPU times limited to 0.3 secondes only shows the efficiency of the procedure.

5.4.2 Example 1 continued.

Models. Timed Event Graph :

$T_1 = 1.11$, $T_2 = 2$, $T_3 = 3.3$, $T_4 = 4$, $T_5 = 4.7$, $T_6 = 3$, $T_7 = 2.5$ and $T_8 = 8$.

P-Time Event Graph :

$T_{10} = 0$, $T_{11} = 9$, $T_{12} = 1$, $T_{13} = 5$, $T_{20} = 5$, $T_{21} = 53$, $T_{22} = 10$ and $T_{23} = 29$.

Off-line preparation.

For $h = 2$, we have

$$D_h = \begin{pmatrix} \varepsilon & \varepsilon & -10 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & -5 & -8 & \varepsilon & \varepsilon \\ 1.11 & 5 & \varepsilon & \varepsilon & -10 & \varepsilon \\ 9 & 3.3 & \varepsilon & \varepsilon & -5 & -8 \\ \varepsilon & \varepsilon & 1.11 & 5 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 9 & 3.3 & \varepsilon & \varepsilon \end{pmatrix}$$

. The application of function star() of Scilab gives $D_h^* =$

$$\begin{pmatrix} 0 & -5 & -10 & -13 & -18 & -21 \\ 1 & 0 & -5 & -8 & -13 & -16 \\ 6 & 5 & 0 & -3 & -8 & -11 \\ 9 & 6 & 1 & 0 & -5 & -8 \\ 14 & 11 & 6 & 5 & 0 & -3 \\ 15 & 14 & 9 & 6 & 1 & 0 \end{pmatrix}.$$

On-line control.

Let $k_s = 1$ and $h = 10$: the horizon is $[k_s, k_s + h] = [1, 11]$ and the problem is the determination of the control u on the horizon $[2, 11]$. The initial state vector $\underline{x}(k_s) = \begin{pmatrix} 0 & 1 \end{pmatrix}^t$.

The simulation is a direct application of the state equality. The analysis of Figure 5.2 which describes output y and desired output $\underline{z}$, shows that condition a) (the process follows the specifications defined by a P-time Event Graph : the number of incoherences of the specifications is null) and condition c) (Just-in-time criteria $y \leq z_m$) are satisfied.

5.5 Kleene star of the block tri-diagonal matrix and formal expressions of the sub-matrices

The determination of the earliest and greatest trajectories needs the calculation of the star of D_h . As this matrix can be large, the CPU time can be important (see Figure 5.3) even if it is polynomial in the strong sense.

Making the most of the specific structure of D_h, we below give another technique which gives the formal expression of the sub-matrices of $(D_h)^*$ using the matrices of the system, that is, A , A^-, $A^=$ and A^+. Therefore, we below show that $(D_h)^*_{j,j} = [w_j \oplus v_{h-j}]^*$ for $j = 0$ to h where

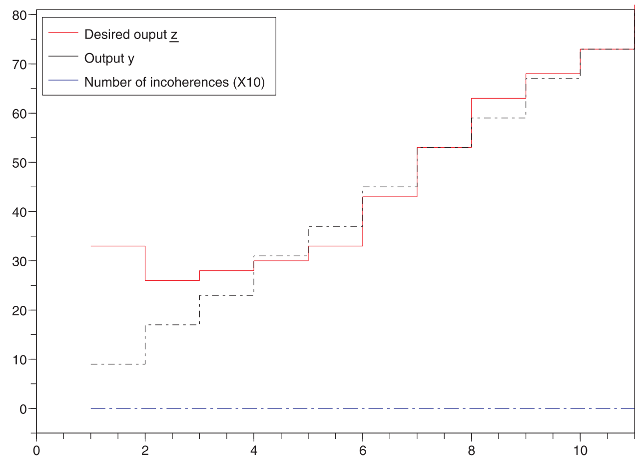

FIG. 5.2 : One step of the simulation

$w_0 = v_0 = A^=$, $w_i = A^= \oplus (A \oplus A^-) \otimes (w_{i-1})^* \otimes A^+$ and $v_i = A^= \oplus A^+ \otimes (v_{i-1})^*(A \oplus A^-)\otimes$ for $i = 1$ to h. Let us present these results.

The star $(D_h)^*$ is the least solution to $Z \geq D_h \otimes Z \oplus I_{(h+1).n}$ and $Z = D_h \otimes Z \oplus I_{(h+1).n}$ where the dimension of matrix Z is $((h+1).n)$ u $((h+1).n)$. We note $Z = \begin{pmatrix} Z_{.,0} & Z_{.,1} & \dots & Z_{.,j} & \dots & Z_{.,h} \end{pmatrix}$ where $Z_{.,j}$ for $j = 0$ to h is a matrix composed of n columns j of Z from $1 + j.n$ to $n + j.n$. The system is a set of $h + 1$ systems whose form is

For $j = 0$ to h, $Z_{.,j} \geq D_h \otimes Z_{.,j} \oplus G_j$ where $G_j = \begin{pmatrix} \varepsilon \\ \dots \\ \varepsilon \\ I_n \\ \varepsilon \\ \dots \\ \varepsilon \end{pmatrix}$, $Z_{.,j} = \begin{pmatrix} Z_{0.,j} \\ \dots \\ Z_{j-1,j} \\ Z_{j,j} \\ Z_{j+1,j} \\ \dots \\ Z_{h,j} \end{pmatrix}$ and the identity matrix I_n of G_j and matrix $Z_{j,j}$ have the same position $j + 1$.

The development of the system is as follows.

$$\begin{cases} Z_{0,j} = A^= \otimes Z_{0,j} \ \oplus A^+ \otimes Z_{1,j} \\ Z_{i,j} = (A \oplus A^-) \otimes Z_{i-1,j} \oplus A^= \otimes Z_{i,j} \ \oplus A^+ \otimes Z_{i+1,j} \\ \text{for } i = 1 \text{ to } j-1 \\ Z_{j,j} = (A \oplus A^-) \otimes Z_{j-1,j} \oplus A^= \otimes Z_{j,j} \oplus A^+ \otimes Z_{j+1,j} \oplus I_n \\ Z_{i,j} = (A \oplus A^-) \otimes Z_{i-1,j} \oplus A^= \otimes Z_{i,j} \ \oplus A^+ \otimes Z_{i+1,j} \\ \text{for } i = \ j+1 \text{ to } h-1 \\ Z_{h,j} = (A \oplus A^-) \otimes Z_{h-1,j} \oplus A^= \otimes Z_{h,j} \end{cases} \tag{5.7}$$

We below consider the least solution of equations and subsystems of (5.7). Note that placing partial least solutions in other equations and subsystems is coherent with the objective of getting the least solution of the complete system since all operations involved are isotone.

Let us consider the following subsystem for $i = 0$ to $j-1$

$$\begin{cases} Z_{0,j} = A^{=} \otimes Z_{0,j} \ \oplus A^{+} \otimes Z_{1,j} \\ Z_{i,j} = (A \oplus A^{-}) \otimes Z_{i-1,j} \oplus A^{=} \otimes Z_{i,j} \ \oplus A^{+} \otimes Z_{i+1,j} \\ \text{ for } i = 1 \text{ to } j-1 \end{cases} \tag{5.8}$$

Let $w_0 = v_0 = A^{=}$, $w_i = A^{=} \oplus (A \oplus A^{-}) \otimes (w_{i-1})^* \otimes A^{+}$ and $v_i = A^{=} \oplus A^{+} \otimes (v_{i-1})^* (A \oplus A^{-}) \otimes$ for $i = 1$ to h.

Property 8. *The least solution to system (5.8) is* $Z_{i,j} = (w_i)^* \otimes A^{+} \otimes Z_{i+1,j}$ *for* $i = 0$ *to* $h-1$ *.*

Proof The proof is based on a recursion. Let $\mathcal{P}(i)$ be defined by $Z_{i,j} = (w_i)^* \otimes A^{+} \otimes Z_{i+1,j}$.

Base case : $\mathcal{P}(0)$

From the first equality of (5.8), we can write $Z_{0,j} = w_0 \otimes Z_{0,j} \oplus A^{+} \otimes Z_{1,j}$ and $Z_{0,j} = (w_0)^* \otimes A^{+} \otimes Z_{1,j}$ where $w_0 = A^{=}$ which proves $\mathcal{P}(0)$. :

Case : $\mathcal{P}(i)$ for i from 1 to $h-1$.

Let us assume $\mathcal{P}(i-1) : Z_{i-1,j} = (w_{i-1})^* \otimes A^{+} \otimes Z_{i,j}$. We will prove that $\mathcal{P}(i-1)$ entails $\mathcal{P}(i)$. As $Z_{i,j} = (A \oplus A^{-}) \otimes Z_{i-1,j} \oplus A^{=} \otimes Z_{i,j} \oplus A^{+} \otimes Z_{i+1,j}$, we deduce : $Z_{i,j} = (A \oplus A^{-}) \otimes (w_{i-1})^* \otimes A^{+} \otimes Z_{i,j} \oplus A^{=} \otimes Z_{i,j} \oplus A^{+} \otimes Z_{i+1,j} =$

$[A^{=} \oplus (A \oplus A^{-}) \otimes (w_{i-1})^* \otimes A^{+}] \otimes Z_{i,j} \oplus A^{+} \otimes Z_{i+1,j} =$

$w_i \otimes Z_{i,j} \oplus A^{+} \otimes Z_{i+1,j}$ with $w_i = A^{=} \oplus (A \oplus A^{-}) \otimes (w_{i-1})^* \otimes A^{+}$. Therefore, $\mathcal{P}(i)$ is deduced from $\mathcal{P}(i-1)$. Moreover, as $\mathcal{P}(0)$ is true, $\mathcal{P}(i)$ has been proved for i from 1 to h : the recursion is finished. ■

Let us consider the following subsystem for $i = j+1$ to h

$$\begin{cases} Z_{i,j} = (A \oplus A^{-}) \otimes Z_{i-1,j} \oplus A^{=} \otimes Z_{i,j} \ \oplus A^{+} \otimes Z_{i+1,j} \\ \text{ for } i = \ j+1 \text{ to } h-1 \\ Z_{h,j} = (A \oplus A^{-}) \otimes Z_{h-1,j} \oplus A^{=} \otimes Z_{h,j} \end{cases} \tag{5.9}$$

Property 9. *The least solution to system (5.9) is* $Z_{i,j} = (v_{h-i})^* \otimes (A \oplus A^{-}) \otimes Z_{i-1,j}$ *for* $i = 1$ *to* h *.*

Proof

The proof is based on a recursion.

$\mathcal{P}(i) : Z_{i,j} = (v_{h-i})^* \otimes (A \oplus A^{-}) \otimes Z_{i-1,j}$.

Base case : $\mathcal{P}(h)$

From the last equality of (5.9), we can write $Z_{h,j} = (A \oplus A^{-}) \otimes Z_{h-1,j} \oplus v_0 \otimes Z_{h,j}$ and $Z_{h,j} = (v_0)^* \otimes (A \oplus A^{-}) \otimes Z_{h-1,j}$ where $v_0 = A^{=}$ which proves $\mathcal{P}(h)$.

Case : $\mathcal{P}(i)$ for i from $h-1$ to 1.

Let us assume $\mathcal{P}(i+1) : Z_{i+1,j} = (v_{h-i-1})^* \otimes (A \oplus A^-) \otimes Z_{i,j}$. We will prove that $\mathcal{P}(i+1)$ entails $\mathcal{P}(i)$. As $Z_{i,j} = (A \oplus A^-) \otimes Z_{i-1,j} \oplus A^= \otimes Z_{i,j} \oplus A^+ \otimes Z_{i+1,j}$, we deduce : $Z_{i,j} = (A \oplus A^-) \otimes Z_{i-1,j} \oplus A^= \otimes Z_{i,j} \oplus A^+ \otimes (v_{h-i-1})^* \otimes (A \oplus A^-) \otimes Z_{i,j} =$

$(A \oplus A^-) \otimes Z_{i-1,j} \oplus [A^= \oplus A^+ \otimes (v_{h-i-1})^* \otimes (A \oplus A^-)] \otimes Z_{i,j} =$

$(A \oplus A^-) \otimes Z_{i-1,j} \oplus v_{h-i} \otimes Z_{i,j}$ with $v_{h-i} = A^= \oplus A^+ \otimes (v_{h-i-1})^* \otimes (A \oplus A^-)$. Therefore, $\mathcal{P}(i)$ is deduced from $\mathcal{P}(i+1)$. Moreover, as $\mathcal{P}(h)$ is true, $\mathcal{P}(i)$ has been proved for i from h to 1 : the recursion is finished.

■

Up to now, the two previous properties cannot calculate the sub-matrices $(D_h)^*_{i,j}$ .The following theorem solves this problem by giving the starting point for each column j.

$(D_h)^*_{i,j} = (w_i)^* \otimes A^+ \otimes (D_h)^*_{i+1,j}$ for $i = 0$ to $j-1$ and

$(D_h)^*_{i,j} = (v_{h-i})^* \otimes (A \oplus A^-) \otimes (D_h)^*_{i-1,j}$ for $i = h$ to $j+1$ with

$(D_h)^*_{j,j} = [w_j \oplus v_{h-j}]^*$ for $j = 0$ to h.

Proof

Properties 8 and 9 give $Z_{j-1,j} = (w_{j-1})^* \otimes A^+ \otimes Z_{j,j}$ and

$Z_{j+1,j} = (v_{h-j-1})^* \otimes (A \oplus A^-) \otimes Z_{j,j}$ which are two specific cases.

Therefore,

$Z_{j,j} = (A \oplus A^-) \otimes Z_{j-1,j} \oplus A^= \otimes Z_{j,j} \oplus A^+ \otimes Z_{j+1,j} \oplus I_n = [(A \oplus A^-) \otimes (w_{j-1})^* \otimes A^+ \oplus A^= \oplus A^+ \otimes (v_{h-j-1})^* \otimes (A \oplus A^-)] \otimes Z_{j,j} \oplus I_n$

Finally, $Z_{j,j} = [(A \oplus A^-) \otimes (w_{j-1})^* \otimes A^+ \oplus A^= \oplus A^+ \otimes (v_{h-j-1})^* \otimes (A \oplus A^-)]^* =$

$[A^= \oplus (A \oplus A^-) \otimes (w_{j-1})^* \otimes A^+ \oplus A^= \oplus A^+ \otimes (v_{h-j-1})^* \otimes (A \oplus A^-)]^* =$

$[w_j \oplus v_{h-j}]^*$which is the sub-matrix (j, j) of $(D_h)^*$.

We can also deduce every matrix of column j of $(D_h)^*$ from properties 8 and 9 :

$(D_h)^*_{i,j} = (w_i)^* \otimes A^+ \otimes (D_h)^*_{i+1,j}$ for $i = 0$ to $j-1$ and

$(D_h)^*_{i,j} = (v_{h-i})^* \otimes (A \oplus A^-) \otimes (D_h)^*_{i-1,j}$ for $i = h$ to $j+1$ ■

We give below $(D_h)^*_{.,j}$ which gives a clear presentation of the results.

$$(D_h)^*_{.,j} = \begin{pmatrix} \dots \\ (D_h)^*_{j-2,j} \\ (D_h)^*_{j-1,j} \\ (D_h)^*_{j,j} \\ (D_h)^*_{j+1,j} \\ (D_h)^*_{j+2,j} \\ \dots \end{pmatrix} =$$

$$\begin{pmatrix} \cdots \\ (w_{j-2})^* \otimes A^+ \otimes (w_{j-1})^* \otimes A^+ \otimes [w_j \oplus v_{h-j}]^* \\ (w_{j-1})^* \otimes A^+ \otimes [w_j \oplus v_{h-j}]^* \\ [w_j \oplus v_{h-j}]^* \\ (v_{h-j-1})^* \otimes (A \oplus A^-) \otimes [w_j \oplus v_{h-j}]^* \\ (v_{h-j-2})^* \otimes (A \oplus A^-) \otimes (v_{h-j-1})^* \otimes (A \oplus A^-) \\ \otimes [w_j \oplus v_{h-j}]^* \\ \cdots \end{pmatrix}$$

Remark 17. $(D_h)^*_{0,0} = (v_h)^*$ *and* $(D_h)^*_{h,h} = (w_h)^*$.

CPU time

We now compare our method with the direct application of function star() of the max-plus toolbox for Scilab 3.1.1 to matrix D_h. Our approach also uses function star() of this max-plus toolbox but for calculating v_i^* and w_i^*. It is known that the complexity of the star is polynomial in the strong sense ($O(n^a)$ with $a = 3$ for the algorithm of Karp which corresponds to $O(h^a.n^a)$ in our problem). The proposed approach makes the most of the specific structure of D_h as its complexity is $O(h.n^a)$.

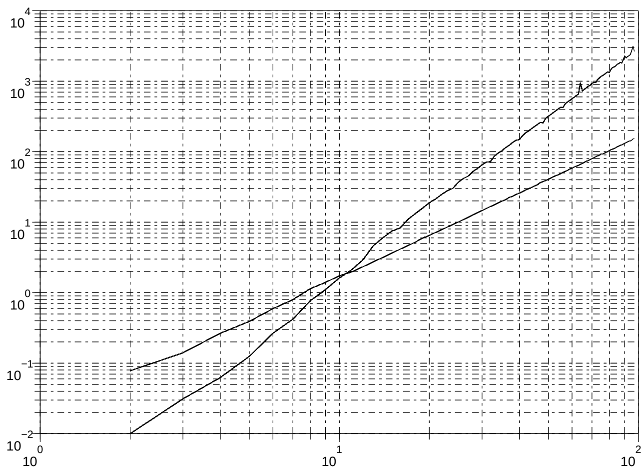

FIG. 5.3 : Off-line preparation : CPU times of the Kleene star of function star() of Scilab and the proposed technique for $h = 50$ and randomly generated matrices

The matrices of the system are randomly generated ($A^= = \varepsilon$, A^- and A^+ are full) and the programme verifies that the two calculated $(D_h)^*$ are identical. Computation tests on CPU time are made with an Intel Core2 Duo 2.26 GHz. We take h =50 and the dimension n of the matrices varies from 2 to 97 : therefore, the considered maximal size of D_h is (4947x 4947). The curves of

Figure 5.3 show a significant improvement of the CPU time : the time gain is about 16 minutes for $n = 80$.

5.6 Conclusion

Contrary to known model predictive control approaches, the numerical on-line control is based on a formal preparation and an important part of the calculations are made off-line in a preparation. In this chapter, we show that this separation between on-line preparation and off-line control where an important part of the calculations is made, allows the consideration of important systems (up to 97 transitions) for long horizons (h =50) which corresponds to the handling of a large scale matrix (4947x 4947). In fact, the calculation of the kleene star of a tri-diagonal matrix can be made once as *it does not depend on the desired output trajectory* : it depends on the models and the size of the given horizon only. In that case, the CPU time of the initial approach (which needs approximately 2.10^3 seconds) is replaced by a new on-line procedure which only needs 0.28 seconds. Therefore, the application field of the model predictive control is expanded : the approach can be applied to long horizons and systems with a reduced CPU time.

A preparation is also made in closed-loop approaches where : the computation of a linear feedback and of the maximal set of the initial states is made off-line ([30]) ; the control is calculated on-line. Recall that approaches based on a feedback defined by a Petri net are limited by the condition that the temporisation and initial marking of each added place are non-negative. Our practical problem is different as the initial condition is uncontrollable which is the usual assumption of the predictive control approaches. Based on the algorithm of [5] whose complexity is doubly exponential, the off-line preparation of [30] can only be made for small systems and an objective is the improvement of the algorithm. Another difficulty is the stabilization of the sequence of semi-modules and some sufficient conditions on the specifications are given (sections III and IV in ([30])). These difficulties are avoided in this chapter as we define a subspace which is an inf-semilattice.

The off-line preparation of the previous approach uses the Kleene star of a special block tri-diagonal matrix denoted D_h which is often encountered in numerical solutions of engineering problems (e.g. computational fluid dynamics, finite element method). Making the most of the specific structure of D_h, we give the formal expression of the sub-matrices of the star of these matrices using the matrices of the system, that is, A , A^-, $A^=$ and A^+. We show that this technique improves the CPU time of the off-line preparation by respect to the direct application of the function star() of Scilab.

Therefore, we have described an approach with a preparation which is close to the closed-loop technique where the corrector is calculated off-line. In a future paper, we will show that a Predictive control using a space corrector can also be made.

References

[1] S. Amari, I. Demongodin, and J.J. Loiseau, 2005. Control of Linear min-plus Systems under Temporal Constraints. *Proc. 44th IEEE Conference on Decision and Control and European Control Conference ECC, CDC-ECC'05.* Seville, Spain, pp.7738-7743.

[2] D. Andersson and S. Vorobyov. *Fast algorithms for monotonic Discounted Linear Programs with two variables per inequality.* preprint NI06019-LAA, Isaac Newton Institute for Mathematical Sciences, Cambridge, United Kingdom, May 2006.

[3] P. Aygalinc, S. Calvez, W. Khansa and S. Collart-Dutilleul, *Using P-time Petri Nets for Robust Control of Manufacturing Systems*, 1st IFAC-Workshop on Manufacturing Systems (MIM'97), Wien, Austria, February 1997, pp. 75-80.

[4] F. Baccelli, G. Cohen, G.J. Olsder J.P. and Quadrat, *Synchronization and Linearity. An Algebra for Discrete Event Systems*, available from http ://maxplus.org, New York, Wiley, 1992.

[5] P. Butkovič and G. Hegedüs. *An elimination method for finding all solutions of the system of linear equations over an extremal algebra.* Ekonomicko-Matematicky Obzor, Vol. 20, no 2, pp. 203-215, 1984.

[6] Y. Cheng and D.-Z. Zheng, *On the cycle Time of Non-autonomous Min-max Systems*, IEEE, 2002.

[7] Y. Cheng and D-Z. Zheng. *Min-max inequalities and the timing verification problem with max and linear constraints.* Discrete Event Dynamic Systems : Theory and Applications, Springer Science and Business Media, Vol 15, pp. 119–143, 2005.

[8] G. Cohen, S. Gaubert, and J.-P. Quadrat, *From first to second-order theory of linear discrete event systems*, 1st SIAC World Congress, Sydney, Australia, 1993.

[9] E. Cohen and Nimrod Megiddo. *Improved algorithms for linear inequalities with two variables per inequality.* SIAM J. Comput., Vol 23, No 6, pp. 1313-1347, December 1994.

[10] R.W. Cottle and A.F. Veinott. *Polyhedral sets having a least element.* Mathematical Programming 3, pp. 238-249, 1972.

[11] J-P. Courtiat, M. Diaz, R.C. De Oliviera and P. Sénac *Formal models for the description of timed behaviors of multimedia and hyper media distributed systems*, Computer Communications 19, 1996, pp. 1134-1150.

[12] B.A. Davey and H.A. Priestley, *Introduction to lattices and order*, Cambridge University Press, Cambridge, UK, 2002.

[13] P. Declerck and M.K Didi Alaoui, *Modeling and liveness analysis of P-time event graphs in the (min, max, +) algebra*, IEEE International Conference on Systems, Man and Cybernetics, The Hague, The Netherlands, October 10-13, 2004.

[14] P. Declerck and M.K. Didi Alaoui, *Optimal Control Synthesis in Interval Descriptor Systems, Application to Time Stream Event Graphs*, IFAC Congress Praha, 2005.

[15] Declerck, P., and Didi Alaoui, M.K. 2005. Extremal trajectories in P-time Event Graphs : application to control synthesis with specifications. *Proc. 44th IEEE Conference on Decision and Control and European Control Conference ECC, CDC-ECC'05.* Seville, Spain, pp.7621-7626.

[16] P. Declerck. *From extremal trajectories to consistency in P-time Event Graphs.* available from http ://www.istia.univ-angers.fr/∼declerck/, technical report, Lisa, University of Angers, September 26, 2007.

[17] P. Declerck, A. Guezzi. *Trajectory Tracking Control of a Timed Event Graph with Specifications Defined by a P-time Event Graph.* POSTA 09 Positive Systems : Theory and Applications, Valencia, Spain, 02-04 September, 2009.

[18] M. Diaz and P. Owezarski *From multimedia models to multimedia transport protocols*, Computer Networks and ISDN Systems 29, 1997, pp. 745-758.

[19] M. Diaz *Les réseaux de Petri : Modèles fondamentaux*, Hermès, Paris, France, 2001.

[20] M.K. Didi Alaoui, P. Declerck *From (min, max, +) algebra to P-time event graphs*, ICINCO'04 International Conference on Informatics in Control, Automation and Robotics Setubal Portugal 2004.

[21] Didi Alaoui, M.K. 2005. Etude et supervision des graphes d'événements temporisés et temporels : vivacité, estimation et commande. *PhD Thesis*, available from http ://www.istia.univ-angers.fr/

[22] S. Collart-Dutilleul and P. Yim, Time window specification and validation with Petri nets, ETFA, IEEE Conference on Emerging Technologies and Factory Automation, September 16-19, 2003, Lisbon, Portugal, pp. 232-237.

[23] S. Gaubert and J. Gunawardena, *the duality theorem for min-max functions* , CRAS, t. 326, Série I, 1998, pp. 43-48.

[24] M. Gondran and M. Minoux. *Graphs and Algorithms.* John Wiley and Sons, March 1984.

[25] M. Gondran and M. Minoux *Graphes, dioïdes et semi-anneaux*, Tec & Doc, Paris, France, 2001.

[26] A. Guezzi, P. Declerck and J.-L. Boimond. *From monotone inequalities to Model Predictive Control.* ETFA 2008, Hamburg, Germany 15-18 September 2008.

[27] S. Hashtrudi Zad, R.H. Kwong and W.M. Wonham, *Supremum operators and computation of supremal elements in system theory*, Siam J. Control Optm., Vol.37, No. 3, 1999, pp. 695-709.

[28] B. Heidergott, G. J. Olsder and L. van der Woude, *Max Plus at Work*, Princeton University Press, 2006.

[29] D. S. Hochbaum. *Monotonizing linear programs with up to two nonzeroes per column.* Operations Research Letters, No. 32, pp 49-58, 2004.

[30] R. D. Katz. *Max-Plus (A,B)-Invariant Spaces and Control of Timed Discrete-Event Systems.* IEEE Transactions on Automatic Control, Vol.52, NO. 2, February 2007, pp. 229-241.

[31] R. Kumar and V.K Garg, *Extremal solutions of inequations over lattices with applications to supervisory control*, Theoret. Comp. Sci., 148, 1995, pp. 67-92.

[32] K. Mc Millan and D. Dill *Algorithms for interface timing verification*, Proceedings of the IEEE, International Conference on Computer Design : VLSI in Computers and Processors, 1992.

[33] V. Pan. *On the complexity of a pivot step of the revised simplex algorithm.* Comp. & Maths with Appls, Vol. 11, N°11, pp.1127-1140, 1985.

[34] M. Queyranne and F. Tardella. *Bimonotone linear inequalities and sublattices of* $\mathbb{R}^n$. Linear Algebra and its Applications 413, pp. 100-120, 2006.

[35] G. Savard. *Introduction aux méthodes de points intérieurs.* Extrait de notes de cours, February 2001.

[36] B. De Schutter and T. van den Boom. *Model predictive control for max-plus-linear discrete event systems.* Automatica, Vol. 37(7), pp. 1049-1056, 2001.

[37] J. Cochet-Terrasson, S. Gaubert and J. Gunawardena, *A constructive fixed point theorem for min-max functions, Dynamics and Stability of Systems*, Vol. 14 N°4, 1999 pp. 407-433.

[38] J. Cochet-Terrasson *Algorithmes d'itération sur les politiques pour les applications monotones contractantes*, thèse, Ecole Nationale Supérieure des Mines de Paris, 2001.

[39] E. Walkup. *Optimization of linear max-plus systems with application to timing analysis* University of Washington, PhD thesis, 1995.

Chapitre 6

Prospective

L'intitulé HDR conduit naturellement à la description d'un projet de recherche scientifique, ce qui est l'objet de cette partie. Cette prospective doit être comprise comme un *projet scientifique idéal* sans limitation de ressources humaines. Bien sûr, un travail de recherche nécessite du temps et de l'énergie, et une première réflexion sur la statégie à suivre face aux contraintes incontournables, sera esquissée dans la dernière partie. Un objectif est d'avoir un projet le plus cohérent possible et les perspectives amenant à une dispersion ne seront pas décrites.

6.1 Evolution passée

- Historiquement, la thématique algèbre des dioïdes (algèbre (max, +) par exemple) existe depuis trois décennies. Elle a été développée au Lisa à Angers depuis plus d'une quinzaine d'années mais aussi dans différents laboratoires en France et à travers le monde. Son principal avantage est son *formalisme* qui permet de présenter clairement des notions complexes et ses possibilités d'analogie avec l'automatique classique mais aussi entre dioïdes.
- Parallèlement à ce mouvement scientifique, la programmation linéaire qui se place dans l'algèbre standard, est beaucoup plus ancienne car on peut remonter au mathématicien Fourier vers 1827. Cependant, la programmation linéaire est toujours l'objet d'une recherche active. Ainsi, une communauté informatique dynamique, placée dans une branche de la programmation linéaire, a donné de *nombreux algorithmes* permettant la résolution de systèmes d'inégalités bivariables depuis au moins trois décennies.
- Enfin, les réseau de Petri inventé par Carl Adam Petri (1962) sont avant tout un modèle graphique décrivant les systèmes à événements discrets avec de nombreux thèmes et domaines d'application. C'est aussi un sujet d'études bien développé en France depuis également trois décennies.

La communauté des réseaux de Petri connaît bien la communauté de l'algèbre (max, +) sans cependant établir des liens scientifiques avancés. La communauté de l'algèbre (max, +) développe des applications en réseaux de Petri. Par contre, ces deux communautés ne semblent pas connaître les résultats scientifiques et les algorithmes de la communauté informatique travaillant sur la résolution de systèmes d'inégalités bivariables.

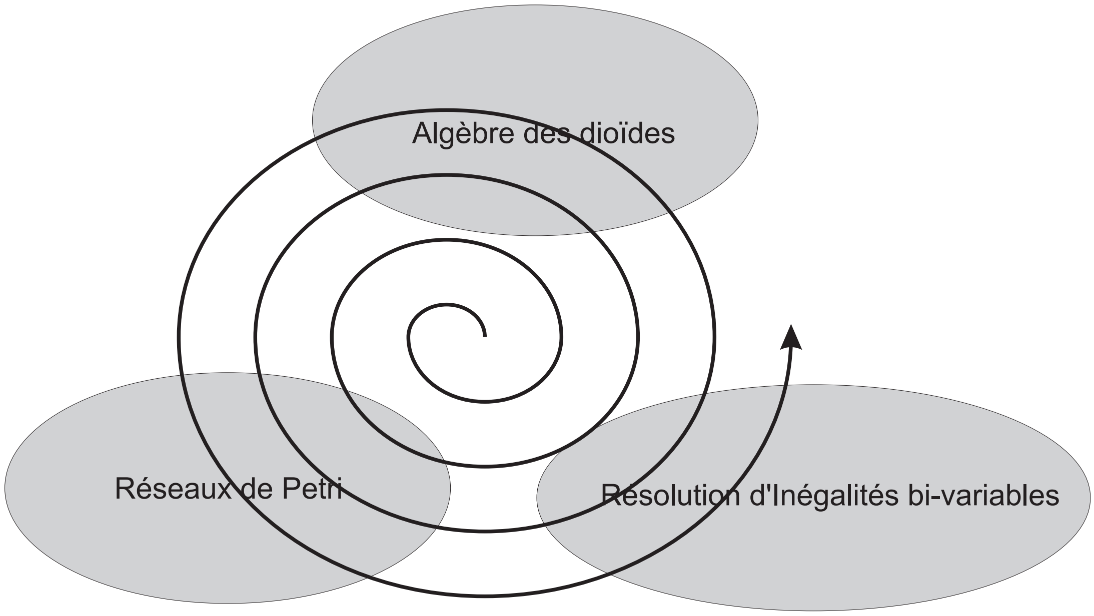

FIG. 6.1 : Trois communautés

6.2 Démarche

Ces 3 communautés travaillent parallèlement alors que leurs travaux présentent des liens forts. Mon projet scientifique a pour objectif essentiel d'intégrer l'algèbre (max, +) et ses variantes dans l'algèbre conventionnelle et de relier les 3 communautés.

Réseaux de Petri / algèbre (max, +). Un premier thème est de développer un langage commun entre la large communauté des réseaux de Petri et la communauté utilisant l'algèbre (max, +) avec pour conséquence un transfert de connaissances entre elles. La technique est de réécrire les travaux (max, +) dans la programmation linéaire tout en gardant les concepts associés aux treillis. Actuellement, peu de chercheurs savent que l'algèbre (max, +) peut être considérée une sous-partie de l'algèbre conventionnelle comme l'illustre la rareté des travaux dans les réseaux de Petri qui sont des contributions à ce sujet.

Communauté informatique / algèbre (max, +). Un second thème est de se relier à une communauté informatique bien développée aux USA et son savoir-faire développé indépendamment de l'algèbre (max, +). L'objectif est de recouper les connaissances et d'exploiter au mieux les apports potentiels. Ce point sera développé par la suite.

Nous présentons ci-dessous quelques étapes dans une logique de gestion de projet.

6.3 Etapes

6.3.1 Première étape exécutée et premières perspectives

Jusqu'à récemment, la communauté travaillant sur l'algèbre (max, +) était séparée de celle importante en France, des réseaux de Petri utilisant souvent l'algèbre conventionnelle. En fait, la thèse de Abdelhak Guezzi soutenue en 2010 (que j'ai encadré avec Jean-Louis Boimond) montre que les problèmes de l'algèbre (max, +) (et dualement l'algèbre (min, +)) peuvent être traités dans l'algèbre conventionnelle moyennant une réécriture du problème sous forme d'une maximisation/minimisation : *nous avons retiré l'opération max (⊕) de l'algèbre (max, +) (*$(\mathbb{R} \cup \{-\infty\}, \oplus, \otimes)$*)* . En particulier, une inéquation d'état dans l'algèbre standard analogue à celle dans l'algèbre (max, +) a été obtenue. On retrouve la commande en juste à temps classique prouvant ainsi qu'une démarche analogue peut complètement être réalisée en exploitant les concepts de treillis. Bien sûr, différentes variations à ce problème peuvent être considérées comme de la poursuite de trajectoires avec spécifications, de l'estimation, ...

De même, le chapitre de ce mémoire sur le temps de cycle d'un graphe d'événements P-temporel généralisé, montre que le temps de cycle ou taux de production est un concept qui ne peut se réduire à l'analyse de circuits simples du graphe d'événements ou d'un graphe associé du système pour les raisons suivantes. Premièrement, la sélection des places est pondérée par un vecteur de réels et pas par un vecteur positif d'entiers avec pour valeurs 0 et 1 : la correspondance matrice/graphe ne peut être que partielle. Deuxièmement, le système ne peut pas être complètement décrit par un graphe associé simple, c'est à dire, présentant un arc standard entre deux sommets, mais par un graphe associé plus complexe. On sort donc d'une théorie des graphes classique et de l'algèbre (max, +) qui ne peut plus s'appliquer directement avec par exemple, le théorème de Karp (qui permet une résolution polynômiale au sens fort). D'autres outils comme la programmation linéaire sont nécessaires afin de traiter ce nouveau type de réseau de Petri. L'existence de ce nouveau système montre qu'un seuil à été franchi et que d'autres systèmes probablement existent. Le problème est alors de les définir, de caractériser leurs modèles mais aussi de les illustrer par des exemples pratiques. L'analyse des fonctions topicales qui englobe les fonctions (min, max, +) présentées dans le chapitre sur la commande est une piste possible (mais ardue) à réfléchir.

Notons également que seule la dépendance entre les lignes a été traitée dans ce chapitre sur le taux, par une multiplication à gauche par un P-invariant généralisé et qu'il reste donc à aussi considérer la dépendance entre colonnes par l'intermédiaire d'une multiplication à droite par un nouveau vecteur, voisin du T-invariant.

6.3.2 Etapes futures avec un point de vue théorique

On se place ci-dessous dans l'espace des réels. Dans cette partie, le terme de complexité doit être pris dans le sens du pire des cas et non en moyenne : l'algorithme du Simplexe mauvais dans le pire cas mais efficace en moyenne dans des cas assez généraux (voir le chapitre 11 du livre de Schriver 1984), ne serra pas considéré. On se focalisera sur la recherche d'une solution optimale ou arbitraire sans considérer le problème de la recherche de l'espace complet, sujet intéressant et bien sûr utile, par exemple en commande pour rester dans un espace.

En programmation linéaire, il est connu que les algorithmes modernes (Karmarkar, Khashiyan,...) sont polynomiaux. En 1984, N. Karmarkar propose la méthode projective. Sa complexité est pseudo-polynomiale en $O(L.n^{3,5})$ (L étant le nombre de bits nécessaires pour représenter les données du problème). Cependant, cette complexité polynomiale est au sens faible (L dans l'expression) ce qui est moins efficace qu'une complexité au sens fort. Il est donc nécessaire de trouver des cas particuliers génériques et des types de problèmes où la résolution peut être réalisée mais avec un algorithme fortement polynomial. On distingue différentes résolutions selon le système de contraintes et le critère à considérer :

- Chaque équation du système linéaire peut avoir 2 variables, 3 et $n > 3$ variables. La difficulté augmente avec le nombre de variables. Notons que des manipulations simples de substitution permettent de transformer un système de $n \geq 3$ variables en un système équivalent avec au plus 3 variables mais pas moins.
- Le problème peut être une optimisation avec un critère linéaire quelconque. Lorsque les coefficients sont du même signe, on peut montrer que le critère revient à une relation d'ordre pour un système monotone (voir Annexe 6.5 ou la thèse de Abdelhak Guezzi) ce qui conduit à une complexité réduite. Enfin, trouver une solution arbitraire est aussi plus aisé en général (ce qui ne veut pas dire que cela soit simple dans l'absolu). En résumé, on peut choisir : de comparer les solutions avec une fonction "objectif" générale ; de comparer les solutions avec une fonction "objectif" particulière pouvant être une relation d'ordre ; de ne pas comparer les solutions.

Suivant ces deux axes, le tableau ci-dessous présentent les différentes résolutions existantes dans la littérature ou à développer. L'axe horizontal correspond aux différents systèmes à traiter, du plus simple au plus complexe en allant de la gauche vers la droite, tandis que l'axe vertical présente les différents types d'objectifs du plus simple au plus compliqué en allant du bas vers le haut. Un système monotone correspond à un type particulier de systèmes voisin mais plus général que le type des inéquations (max, $+$) (voir Annexe 6.5) qui permet d'avoir des solutions extrémales uniques. Le terme simplement (respectivement, fortement) correspond à une complexité simplement polynomial (respectivement, "fortement polynomial").

	2 variables et système monotone	2 variables	3 variables avec hypothèses	Cas général : n variables ≥ 3
Optimisation générale : $c$quelconque	(4)			simplement polynomial
$c > 0$ ou $c < 0$	fortement polynomial (2)	(5)		simplement polynomial
Détermination d'une solution arbitraire	fortement polynomial (1)	fortement polynomial (3)	(6)	simplement polynomial

Le problème est d'analyser les différents cas et de développer les algorithmes correspondants.

Nous avons montré que le point (2) (et (1) a fortiori) correspond à l'algèbre (max, $+$) qui exploite le concept de treillis (voir théorème 4.9 page 94 dans la thèse de A. Guezzi). Les algorithmes correspondants sont performants ($O(n^2)$, $O(n^3)$) car ils correspondent à la théorie des graphes classique (le livre de Gondran et Minoux est la référence en ce domaine). Le point (3) peut être résolu par un algorithme fortement polynomial ($O(m.n^2.\log(m)$) Dorit S. Hochbaum de l'université de Berkeley USA en 1993, $O(m.n^2.(\log(m)+\log^2(n)))$ Cohen et Megiddo en 1993). Ainsi, D.S. Hochbaum moyennant la réécriture du système sous une forme monotone revient à la résolution précédente qui est efficace.

A l'inverse, la colonne de droite est une autre zone qui représente les cas simplement polynomiaux, numériquement moins performants mais dans un cadre général plus complexe.

A partir de ces deux zones, on peut déduire du tableau que la zone de cas où la recherche est innovante est entre les deux et en particulier, dans la diagonale (4), (5) et (6). A notre connaissance, il n'existe pas d'algorithmes spécifiques aux cas (4) et (5).

Certains travaux existent pour le cas (6) moyennant l'ajout d'hypothèses restrictives : dans une publication de D.S. Hochbaum, une inégalité peut avoir 3 variables mais une variable n'apparaît que dans cette relation. Le problème traité par Karp présente une correspondance avec 3 variables où une variable λ est commune à toutes les inéquations moyennant une simplification ; l'algorithme est fortement polynomiale ($O(n^3)$).

Extension du tableau

Le tableau précédent propose un schéma de pensée et un survol commode de différents problèmes. Il n'est pas limitatif et d'autres questions peuvent être posées.

Notons ainsi que la considération non d'inégalités mais d'égalités (max, $+$) conduit à l'écriture d'une égalité particulière supplémentaire qui est appelée la condition complémentaire peu simple à traiter (produit de différences). Les polyèdres complémentés par cette relation a été l'objet de travaux mathématiques depuis les années 70.

D'autre part, ce tableau ne recouvre pas tous les dioïdes. Une question est de déterminer comment des équations et inéquations formulées dans un dioïde différent de l'algèbre (max, $+$) se réécrivent dans l'algèbre standard. Ce travail méticuleux peut permettre un transfert intéressant de connaissances d'un domaine dans l'autre.

Notons enfin que le tableau précédent fait abstraction du caractère entier qui rajoute une dimension parfois non triviale aux différents problèmes.

6.3.3 Remarque sur la mise en oeuvre

Le tableau précédent et ses extensions possibles, se place dans un objectif global de généralisation de l'algèbre (max, +) ou (min, +). Son exploitation demande un réel travail difficile à estimer. Mettant en valeur de nombreux problèmes, il exige sans doute des choix de questions à traiter et à ne pas traiter, malgré un désir naturel "de tout faire". Son exécution partielle et sans précipitation est le garant de l'établissement de résultats fiables. Considéré ci-dessous, un autre point de vue est "d'ajuster" les problèmes aux outils théoriques disponibles, ce qui demande un investissement également certain.

6.3.4 Etapes futures avec un point de vue inverse

Comme l'illustre le cas (1) de l'algèbre (max, +), le tableau recouvre de nombreuses applications. Un objectif peut être également de trouver non la technique de résolution pour un cas donné ce qui peut demander des développements mathématiques avancés, mais la correspondance entre un cas et un problème théorique avec application à résoudre, ce qui est d'avantage dans les domaines d'activités de la section 61. Bien sûr, les points de vue théorique et inverse ne sont pas exclusifs et peuvent s'enrichir mutuellement.

Dans ce point de vue inverse, nous citons ci-dessous quelques remarques et perspectives à court terme. Les outils principaux sont ceux de la programmation linéaire et les treillis. La stratégie va être, soit d'exploiter la propriété de système monotone sur un nouveau système dans lequel on a repéré cette propriété, soit de *construire* cette propriété quand le système ne présente pas cette propriété. On pourra alors appliquer la théorie existante qui a fait ses preuves et utiliser les algorithmes efficaces déjà développés.

- Le chapitre de ce mémoire sur l'évaluation de performance (temps de cycle) donne un exemple où chaque inéquation présente 4 variables dont une, la variable λ est commune à toutes les inéquations. Ceci est plus complexe que le cas (6) avec 3 variables avec hypothèses. Ce système n'est pas monotone en raison de la dépendance entre tâches (voir quatrième ligne de A_{11}). Il en est de même pour les matrices définissant le processus (voir matrice G^- et sa quatrième ligne avec quatre coefficients non nuls). Afin de pouvoir appliquer un algorithme spécialisé performant et ne pas appliquer un algorithme générique moins efficace, une piste intéressante est de re-modéliser le processus dans l'objectif d'obtenir un problème plus simple correspondant au cas (6) où des outils théoriques existent.

- Le chapitre sur la commande prédictive a mis en valeur la nécessité d'algorithmes efficaces. La gestion des retards de la procédure est une question pratique qui peut être traitée de la manière suivante. Plus précisément, lorsqu'une commande en juste à temps est sans solution, c'est à dire que la trajectoire de sortie n'est pas plus petite qu'une trajectoire désirée, on peut essayer d'accepter un retard qu'il faudra minimiser autant qu'il est possible. Le problème reformulé avec ce retard, est alors non-monotone. Dans cette situation, nos travaux en cours (encore à l'état de brouillon) montrent que le cas (3) (2 variables et pas de propriété de système monotone) peut être appliqué aux réseaux de Petri pour trouver une trajectoire arbitraire. La technique est alors de construire un nouveau système, monotone, qui est une image du système initial. Le nouveau système sera le réseau de Petri initial auquel sera joint un double mais avec les arcs inversés et un marquage négatif. Une solution arbitraire pour ce système virtuel et un critère "simple", d'ordre composante par composante, pourra être calculée rapidement. Elle serra alors le point de départ de la résolution d'un algorithme générique suivant un critère complexe. La procédure permettra de générer rapidement une solution arbitraire ce qui augmentera la robustesse de la procédure.

- Dualement à l'approche dateur où un polyèdre peut modéliser différents réseaux de Petri comme le Graphe d'Evénements Temporisé, le Graphe d'Evénements P-temporels avec dépendances entre tâches, il est possible d'avoir une approche compteur et de modéliser des processus sous forme d'un polyèdre $A.x \leq b$ où x est un vecteur de comptes, A exprime la structure et b est le marquage initial des places. Le marquage courant peut alors être directement déduit avec l'équation fondamentale du marquage connaissant x. L'intérêt essentiel de ce modèle est qu'il peut aisément décrire des structures complexes de réseaux de Petri où les places peuvent avoir plusieurs arcs entrants et sortants contrairement au graphe d'événements. Cette modélisation est aussi complète dans le sens que toute l'information se trouve dans le modèle algèbrique et qu'il n'est plus nécessaire de considérer le réseau de Petri, passé cette étape, à part pour illustrer les résultats. On peut aussi tenir compte des valuations sur les arcs. Rappelons cependant que ce type de modèle compteur ne pourra pas être aussi efficace pour modéliser des phénomènes complexes liées aux temps comme les synchronisations complexes d'un graphe d'Evénements P-temporel (voir chapitre sur la vivacité), d'un graphe d'Evénements à flux (voir chapitre sur la commande) et les dépendances de tâches (voir chapitre sur le temps de cycle) entre autres. Bien sûr, on pourra toujours accepter un modèle incohérent qui nécessitera des traitements informatiques lourds et peu efficaces et bloquera le développement d'une théorie élégante. Sur ce modèle compteur, différents problèmes pratiques peuvent alors être envisagés. On peut ainsi faire de l'estimation de séquence de compteur et de marquage sur un horizon glissant connaissant l'occurence de tir de certaines transitions et moyennant l'hypothèse d'absence de conflit rétroactif. Cette étude est actuellement développée avec Patrice Bonhomme de l'Université de Tours. Vu l'ampleur des travaux orientés sur les réseaux de Petri dits ordinaires, on peut soupçonner l'existence de nombreuses problèmatiques intéressantes comme l'atteignabilité d'un marquage, la détection de défaillances,...

Egalement notons qu'une équation d'état discrétisée des systèmes continus peut être réécrite facilement sous la forme d'une inégalité ce qui est une forme cohérente avec les inéquations d'état obtenus pour les systèmes à événements discrets. On peut donc a priori réfléchir, dans une perspective à long terme, sur un modèle hybride mixant systèmes continus/ systèmes à événements discrets en tenant compte du repérage différent de l'état dans le temps et les événements.

6.4 Potentialités

Ce projet scientifique qui se place dans un contexte ne présentant aucune limitation de ressources humaines, s'appuie sur trois bases scientifiques complémentaires. Il peut intéresser les laboratoires qui ont un savoir- faire dans l'algèbre (max, +) et les treillis, les équipes en réseaux de Petri, mais aussi les équipes travaillant sur la complexité des algorithmes, thème connu en informatique. Si on se concentre exclusivement sur la conception d'algorithmes efficaces, la réalisation de ce projet est naturellement ambitieuse car elle nécessite une équipe de chercheurs en automatique et informatique théorique. Sur un point de l'histoire des sciences, la rareté des années de références marquantes où de réelles innovations théoriques ont été réalisées (voir le chapitre 11.8 dans [6] et en particulier, les années de création des théorèmes de Zermelo (1908), Zorn (1935), Bourbaki (1940), Kneser (1950), Tarski (1955) et Amann (1977) sur la notion d'ordre et le point fixe au cours du vingtième siècle), est un appel au réalisme. Elle montre aussi que ce projet se place dans le long terme même si trouver des applications possibles à des outils théoriques existants et les mettre en oeuvre, est dans le court et moyen terme.

−−−−−−−−−−−−−−−−−−−−−−−−−

6.5 Annexe : inégalité monotone et élément extremum

Dans cette partie, nous considérons l'ordre partiel défini sur un ensemble S qui est défini comme suit : $x \leq y \quad \Leftrightarrow x_i \leq y_i \; \forall i \in \{1, 2, ..., card(x)\}$. Nous rappelons le vocabulaire des treillis [1] [6] afin de faciliter la lecture du document : ceci sera appliqué à une classe particulière des systèmes linéaires. Un maximum (respectivement, minimum) d'un sous-ensemble est un élément de cet ensemble qui est plus grand (respectivement, plus petit) que tout autre élément du sous-ensemble. S'il existe, il est unique.

Un majorant (respectivement, minorant) d'un sous-ensemble est un élément pas nécessairement appartenant à ce dernier qui est le plus grand (respectivement, plus petit) que tout autre élément du sous-ensemble. Si un majorant appartient au sous-ensemble, c'est le maximum (respectivement, minimum) élément. Le majorant (respectivement, minorant) est aussi appelé la borne supérieure (respectivement, inférieure).

Lorsque l'ensemble des majorants ou bornes supérieures a un plus petit élément, ce dernier est appelé plus petite borne supérieure. De même lorsque l'ensemble des minorants ou bornes inférieures a un plus grand élément, on l'appelle plus grande borne inférieure. Un ensemble ordonné $(S, \leq)$ est un sup-demi-treillis (respectivement, inf-demi-treillis) si toute partie de deux éléments de S admet une plus petite borne supérieure (plus grande borne inférieure). Il est appelé treillis si il est à la fois, un inf-demi-treillis et un sup-demi-treillis.

L'intersection d'un nombre fini de demi-plans dans $\mathbb{R}^n$ est appelé un polyèdre convexe ou simplement un polyèdre. Un polyèdre est un sous-ensemble Γ de $\mathbb{R}^n$ qui peut être représenté comme la solution d'un système d'inégalités linéaires : $\Gamma = \{x \in \mathbb{R}^n \colon Ax \leq b\}$ où A est une matrice réelle $m \times n$ et b est un vecteur de dimension m.

Dans cette partie nous nous concentrons sur une classe particulière des systèmes linéaires définis comme suit :

Définition 6.1. *Le système des inégalités linéaires $Ax \leq b$ est inf-monotone (respectivement, sup-monotone) si chaque ligne de la matrice A a un coefficient strictement négatif (respectivement positif) au plus.*

Définition 6.2. *Si en même temps, l'inégalité est inf-monotone et sup-monotone, cette inégalité est dite bi-monotone. Un système est appelé bi-monotone si chaque inégalité est bi-monotone.*

Les concepts de treillis et de système monotone sont maintenant reliés. Le maximum (respectivement minimum) d'un couple $\{x, y\}$ est noté $x \vee y$ (respectivement, $x \wedge y$) .

Théorème 6.3. *[2], Theorem 1 in [3] Soit Γ l'ensemble des solutions d'un système inf-monotone (respectivement sup-monotone) $Ax \leq b$. Si x et y sont deux éléments de Γ alors leur minimum $x \wedge y$ (respectivement, maximum $x \vee y$) appartient à Γ.* ■

Corollaire 6.4. *L'ensemble* Γ *d'un système inf-monotone (respectivement, sup-monotone) est un inf-demi-treillis (respectivement sup-demi-treillis).* ■

Corollaire 6.5. *L'ensemble* Γ *d'un système bi-monotone* $Ax \leq b$ *est un treillis.* ■

Le théorème suivant garantit l'existence d'une solution minimale ou maximale d'un ensemble Γ.

Théorème 6.6. *Corollary 2 in [2], [3]. L'ensemble* Γ *d'un système inf-monotone (respectivement sup-monotone) a un plus petit (respectivement, plus grand) élément si l'ensemble est non vide et a un minorant (respectivement, majorant).* ■

Le résultat suivant fait le lien entre la fonction objectif de la programmation linéaire et l'ordre composante par composante. Le résultat suivant est brièvement déclaré dans [2] [3].

Théorème 6.7. *- Les propositions suivantes sont équivalentes :*
1. L'ensemble $\Gamma = \{x \in \mathbb{R}^n : Ax \leq b\}$ *a un élément maximal* x_0
2. x_0 *est optimal pour le problème* $max\{cx,$ *tel que* $Ax \leq b$ *pour tout* $c > 0\}$ *.*
- De plus, l'équivalence précédente entre les points 1 et 2 est vérifiée, si les mots maximal et max, sont remplacés par, minimal et min. ■

En conséquence, les approches utilisant la programmation linéaire pour $c > 0$ obéissent à une relation d'ordre composante par composante si le système a un maximum ou un minimum. Ainsi, les algorithmes de la programmation linéaire peuvent aussi résoudre les problèmes utilisant la relation d'ordre.

Bibliographie

[1] F. Baccelli, G. Cohen, G.J. Olsder and J.P. Quadrat, Synchronization and Linearity. An Algebra for Discrete Event Systems, available from http ://maxplus.org, New York, Wiley, 1992.

[2] Cottle, R.W. and A.F. Veinott, *Polyhedral sets having a least element*, Mathematical Programming, Vol. 3, pages : 238-249, 1972.

[3] Gabbay, H. *A note on polyhedral sets having a least element*, Mathematical Programming, Vol. 11, Pages : 94-96, 1976.

[4] Queyranne, M. and F. Tardella, *Bimonotone linear inequalities and sublattices of* R^n, Linear Algebra and its Applications, Vol. 413, Pages : 100-120, 2006 .

[5] Veinott A. , *Extreme Points of Leontief Substitution Systems*, Linear Algebra and its Applications, Vol. 1, Pages : 181-194, 1968.

[6] E. Zeidler, Nonlinear Functional Analysis and its Applications I. Fixed-Point Theorems, Springer-Verlag New York, Inc., 1986.

5. Liste des titres et travaux

Ouvrage collectif

- Ph. Declerck et A. Guezzi

Lecture Notes in Control and Information Sciences 389, Trajectory Tracking Control of a Timed Event Graph with Specifications Defined by a P-time Event Graph, (POSTA 09 Positive Systems: Theory and Applications, Valence, Espagne, 2-4 septembre 2009) Springer, Rafael Bru, Sergio Romero-Vivo (Eds.), pages 279-290, 2009.

Revues internationales à comité de lecture

- Ph. Declerck

From extremal trajectories to consistency in P-time Event Graphs, IEEE Transactions on Automatic Control, Vol. 56 n°2, IETAA9, pages 463-467, février 2011.

- Ph. Declerck et M.K. Didi Alaoui,

Optimal control synthesis of timed event graphs with interval model specifications, IEEE Transactions on Automatic Control, vol.55, n°2, IETAA9 , pages 518-523, février 2010.

- A. Guezzi, Ph. Declerck et J.-L. Boimond

Commande de graphes d'événements temporisés sur un horizon glissant, Journal Européen des Systèmes Automatisés (JESA), vol. 43/7-9, pages 1097-1111, accessible sur http://jesa.revuesonline.com/acceuil.jsp, 2009.

- Ph. Declerck

Predictability and control synthesis in Timed Event Graphs. Kybernetika, Académie des Sciences de la République Tchèque, vol. 35, n°1, pages 25-38, disponible sur http://www.kybernetika.cz/, 1999.

Conférences internationales

- Ph. Declerck

Trajectory Tracking Control of a Timed Event Graph with Specifications Defined by a P-time Event Graph: On-line control and Off-line preparation, Wodes 2010, pages 165-170, CD-ROM, Berlin, Allemagne, 30 août au 1 septembre, 2010.

- A. Guezzi et Ph. Declerck

From Linear Programming to Graph Theory: Standardization of the Algebraic Model of Timed Event Graphs, CIE39, International Conference on Computers & Industrial Engineering, CD-ROM, Troyes, France, 6-8 juillet, 2009.

- A. Guezzi et Ph. Declerck

Performance evaluation of discret event systems using P-time Event Graphs, ETFA 09 14th IEEE International Conference on Emerging Technologies and Factory, CD-ROM, Mallorca, Espagne, 22-26 septembre, 2009.

- A. Guezzi, Ph. Declerck et J.-L. Boimond

From monotone inequalities to Model Predictive Control, ETFA 2008, CD-ROM, Hamburg, Germany 15-18 septembre, 2008.

- Ph. Declerck, A. Guezzi et C. Gros

Temps de cycle des Graphes d'Événements Temporisés et P-temporels, CIFA 08, CD-ROM, Bucarest, Roumanie, 3-5 septembre, 2008.

- Ph. Declerck, A. Guezzi et J.-L. Boimond

Cycle Time of P-time Event Graphs, 4th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2007), Special Session: From Planning to Control of Manufacturing Systems, CD-ROM, Angers, France, 09-12 mai, 2007.

- Ph. Declerck

Detection of changes by Observer in Timed Event Graphs and Time Stream Event Graphs, IFAC DCDS'07, pages 215-220, Cachan, 13-15 juin, 2007.

- Ph. Declerck et K. Didi Alaoui

Liveness and acceptable trajectories in P-time Event Graphs, Wodes 2006, pages 445-450, Ann Arbor, Michigan, USA, 10-12 juillet, 2006.

- Ph. Declerck et K. Didi Alaoui

Extremal trajectories in P-time Event Graphs. Application to control synthesis with specifications. CDC-ECC'05 , 44th IEEE Conference on Decision and Control and European Control Conference, ECC 2005, CD-ROM, Séville, Espagne, 12-15 décembre 2005.

- Ph. Declerck et K. Didi Alaoui

Optimal Control Synthesis in Interval Descriptor Systems. Application to Time Stream Event Graphs, 16th IFAC World Congress, CD-ROM, Prague, 3-8 juillet 2005.

- Ph. Declerck et K. Didi Alaoui

State Estimation in Time Event Graphs. Application to fault detection, 17th IMACS World Congress, Scientific Computation, Applied Mathematics and Simulation, CD-ROM, Paris, France, 11 15 juillet, 2005.

- I. Ouerghi, Ph. Declerck et J.L. Ferrier

Estimation des durées algébriques entre les événements d'un graphe IPERT , Conférence Internationale Francophone d'Automatique, CIFA 2004, CD-ROM, Douz Tunisie, 22- 24 novembre, 2004.

- K. Didi Alaoui et Ph. Declerck

From (min, max, +) algebra to P-time event graphs, ICINCO 2004 International Conference on Informatics in Control, Automation and Robotics, CD-ROM, Setubal Portugal, 25-28 août, 2004.

- Ph. Declerck et K. Didi Alaoui

Modelling and liveness analysis of P-time event graphs in the (min, max, +) algebra, SMC'2004, IEEE International Conference on Systems, Man & Cybernetics, CD-ROM, The Hague, The Netherlands, 10- 13 octobre, 2004.

- Ph. Declerck et M.K. Didi Alaoui

From constraint propagation to control synthesis and state estimation, International Workshop on Max-algebra (IWMA), The University of Birmingham, School of Mathematics and Statistics, 30 juin - 3 juillet, 2003.

- M.K. Didi Alaoui et Ph. Declerck

Control synthesis in interval systems, Automated Manufacturing Systems and Development Methodology, The 9th IEEE International Conference on Emerging, ETFA2003, Volume I pages 634-639, Lisbon, Portugal, 16-19 septembre, 2003.

- Ph. Declerck et M.K. Didi Alaoui

Constraint propagation for max-plus-linear discrete event systems: application to the state estimation, Symposium: Discrete Events in Industrial and Manufacturing Systems, The IMACS/IEEE Multiconference CESA' 2003, Lille, France, 9-11 juillet, 2003.

- Ph. Declerck, M.K. Didi Alaoui et M. Bourcerie

Estimation d'état par l'étoile de Kleene, CIFA 2002, Conférence Internationale Francophone d'Automatique, CD-ROM, Nantes, du 8 au 10 juillet, 2002.

- Ph. Declerck

Estimation, prediction and control in (max,+) systems, first IFAC Symposium on System Structure and Control, workshop on max-plus algebras, Pages 139-144, CD-ROM, Praha, 27-29 août, 2001.

- Ph. Declerck et R. Guihur

Supervisory control after perturbation, SMC 2000, IEEE International Conference on Systems, Man, and Cybernetics, CD-ROM, Nashville, USA, 8-11 Octobre, 2000.

- Ph. Declerck

Estimation, Prédiction et Commande après Rupture de Modèle, Conférence Internationale Francophone d'Automatique, première manifestation IEEE francophone, CIFA 2000, pages 326-330, Lille, 5 au 8 juillet, 2000.

- Ph. Declerck et R. Guihur.

General predictor in the algebra of dioïds, ETFA99, 7th IEEE International Conference on Emerging Technologies and Factory Automation, pages 1057-1062, 18-22 octobre, 1999.

- Ph. Declerck

Predictability and Control synthesis in Timed Event Graphs in Timed Event Graphs, 4th Workshop on Discrete Event Systems, pages 475-480, Cagliari, Italie, 26-28 août, 1998.

- Ph. Declerck et M. Mares

Temporal Control synthesis and Failure Recovery, 9th Symposium of the IFAC on Information Control in Manufacturing (INCOM'98), pages 385-390, Nancy-Metz, 24-26 juin,1998.

- Ph. Declerck

Control synthesis using the state equations and the "ARMA" model in Timed Event

Graphs, 5th IEEE Mediterranean Conference on Control and Systems, session invité, CD-ROM, Cyprus, 21-23 juillet, 1997.

- Ph. Declerck
Multi-step Approach for Control synthesis using the "ARMA" model in Timed Event Graphs, 15th IMACS World Congress on Scientific Computation, Modelling and Applied Mathematics, Berlin, 24-29 août, 1997.

- Ph. Declerck
"ARMA" model and admissible trajectories in Timed Event Graphs, CESA'96, IMACS, IEEE-SMC, pages 669-674, Lille, France, 9-12 juillet, 1996.

- Ph. Declerck et M.Staroswiecki
Generation of analytical redundancy relations for fault detection, International conference Sicici' 92, IEEE, Singapore, 18-21 février, 1992.

- J-Ph.Cassar , M.Staroswiecki et Ph.Declerck
Structural decomposition of large scale system for the design of failure detection and isolation procedures, Systems Sciences 11, Wroclaw, septembre, 1992.

- Ph. Declerck et M.Staroswiecki
Identification of structurally solvable sub-systems for the design of Fault Detection and Isolation Schemes, using the Embedding Procedure, 9th IFAC/IFORS Symposium on Identification and System Parameter Estimation, pages 230-235, Vol.1, Budapest, Hongrie, 8-12 juillet, 1991.

- Ph. Declerck et M.Staroswiecki
Characterization of the Canonical Components of a Structural Graph for Fault Detection in Large Scale Industrial Plants, First European Control Conference, ECC'91, pages 298-303, Vol.1, Grenoble, France, 2-5 juillet, 1991.

- M.Staroswiecki et Ph.Declerck

Analytical redundancy in non linear interconnected systems by means of structural analysis, IFAC Symposium on Advanced Information Processing in Automatic Control, Nancy, 3-5 juillet, 1989.

- M.Staroswiecki, J-Ph.Cassar et Ph.Declerck
Automatic Analytical Redundancy relationships generation in complex interconnected system based on a structural approach, INCARF 89, New Delhi, 1989.

Manifestations nationales

- A. Guezzi, Ph. Declerck et J.-L. Boimond, Commande de graphes d'évènements temporisés par les systèmes monotones, JD MACS09 : 3èmes Journées Doctorales du GdR MACS, Angers, France, 17-18 mars, 2009.

- Ph. Declerck et M.K. Didi Alaoui
Synthèse de commande des systèmes dynamiques à événements discrets, quatrième Colloque Francophone sur la Modélisation des Systèmes Réactifs (MSR), pages 179-194, Metz, 6 au 8 octobre, 2003.

- Ph. Declerck et R. Guihur.
Transitoires et comportements périodiques, 2ème Congrès sur la Modélisation des Systèmes réactifs, MSR'99, pages 39-48, Cachan, 24-25 mars, 1999.

Séminaires

- A. Guezzi,
Temps de cycle des Graphes d'Événements Temporisés et P-temporels, GDR, Paris, 22 janvier, 2009.

- A. Guezzi, Ph. Declerck et J.-L. Boimond.

Cycle Time of P-time Event Graphs, GDR MACS, Journées Sciences et Techniques de la Production de Biens et de Services, Aix-en-Provence, 21-23 novembre, 2007.

- M.K. Didi Alaoui et Ph. Declerck

Estimation d'état pour les modèles d'intervalles, MEA (méthodes ensemblistes pour l'automatique), Paris, 2005.

- M.K. Didi Alaoui et Ph. Declerck

Commande et estimation d'état par l'algèbre (max, +), 8ème Atelier d'évaluation de performances, Reims, 12 au 16 mai, 2003.

Rapports industriels

- M.Staroswiecki et Ph. Declerck

Rapports de contrat EDF phases I, II et III : Méthode et Modèle de Validation Fonctionnelle pour la Régulation de Niveau du Générateur de Vapeur des REP 900 MW, Juin 89.

Mémoires de thèse

- Abdelhak Guezzi.

Modélisation, analyse de performance et commande des systèmes à événements discrets. Cotutelle 80% avec Jean-Louis Boimond 20%, 2006/2010. Thèse soutenue le 24 septembre 2010 à l'Université d'Angers.

- Mohamed Khalid Didi Alaoui.

Etude et supervision des graphes d'événements temporisés et temporels : vivacité, estimation et commande. Cotutelle avec Marc Bourcerie. Thèse soutenue le 7 décembre 2005 à l'Université d'Angers.

- Ph. Declerck.

Analyse structurale et fonctionnelle des grands systèmes. Application à une centrale PWR

900 MW. Thèse soutenue le 20 décembre 91 à l'Université de Lille I.

Philippe Declerck

Predictability and control synthesis

Kybernetika, Vol. 35 (1999), No. 1, [25]--38

Persistent URL: http://dml.cz/dmlcz/135265

KYBERNETIKA — VOLUME 35 (1999), NUMBER 1, PAGES 25-38

PREDICTABILITY AND CONTROL SYNTHESIS

PHILIPPE DECLERCK

Processes modeled by a timed event graph may be represented by a linear model in dioïd algebra. The aim of this paper is to make temporal control synthesis when state vector is unknown. This information loss is compensated by the use of a simple model, the "ARMA" equations, which enables to introduce the concept of predictability. The comparison of the predictable output trajectory with the desired output determines the reachability of the objective.

1. INTRODUCTION

Discrete Event Dynamic Systems (DEDS) represent a great number of systems such as flexible manufacturing systems, multiprocessor systems, and transportation networks that are characterized as being concurrent, asynchronous, distributed and parallel. Among formalisms used to represent DEDS, Timed Petri Nets explicitly integrate time. Timed Event Graphs are a subclass that plays an important role because of its deterministic behavior. Its evolution is described by linear systems defined on a dioïd. The interpretation of each variable is, for example, of "dater" type for (max,+) algebra: each function $x_i(k)$ represents the date of the kth firing of transition x_i; $\oplus$ stands for the max operation while the usual addition plays the role of the multiplication, denoted $\otimes$.

An important objective is to make temporal control synthesis of systems. Historically, the PERT graph and potentials-tasks are the first well-known classical approaches enabling the definition of the execution calendar of a given project [13]. The results can be given by two algorithms that give respectively the earliest times and the latest times of the tasks. Using the dioïd algebra, [1, 7] generalize for processes with repetitive tasks. They solve the following classical problem: given a production system, how can we compute the latest dates of the part inputs in such a way that the parts be produced *at the latest before* the desired dates? It can be proved that, for the system which dater equations gives the lowest solution (the earliest times), the greatest solution (the latest times) is explicitly given by the backward recursive equations where the co-state vector plays the role of the state vector. This control theory is similar to the adjoint-state equations of optimal control theory. The difference between the co-state and the state represent the "spare time" or the "margin" which is available for the firing of the transitions. A negative difference prevents the future deadlines from being met.

Thus, this approach requires the vector state values. The knowledge of the model and of the initial conditions enable us to characterize the state vector with a state equation iteration. Unfortunately, this solution disregards unavoidable model errors and must start from a known state. To overtake these difficulties, we propose the use of a different model composed of equations called "ARMA" by analogy with ARMA equations used in classical control system theory. We show the possibility of using "ARMA" model to make a temporal control synthesis without knowing the state vector [8]. By example, this situation occurs when the process undergoes a failure and must be recovered. The model presents a description rupture, generating a misappreciation of the state vector. In this case, the problem is to compute, *after this past evolution of the system*, the latest firing dates of the input transitions in such a way that the output events occur at the latest before the desired date [9]. The model is supposed to be exact in the horizon of application of the control synthesis [10].

This paper is organized as follows. We first give notations and background concerning dioïds. We then, present the problem and study the "predictability" concept for the "d-cyclic" systems. Finally, we propose a multi-step control synthesis based on the "ARMA" model. The approach is applied to a short example in the annex.

2. PRELIMINARY

One of the tools used in this paper is (max,+) algebra, a particular example of the algebraic structure generally called dioïd. In this introduction, we shall limit ourselves to present notations and main concepts. A complete description may be found in [1][11].

A semi-ring S is a triplet $(S, \oplus, \otimes)$ where $(S, \oplus)$ and $(S, \otimes)$ are monoïds, $\oplus$ is commutative, $\otimes$ is distributive with respect to $\oplus$ and the zero element of $\oplus$ is the absorbing element of $\otimes$. A dioïd D is an idempotent semi-ring. The set $\Re \cup \{-\infty\}$ provided with max denoted $\oplus$ and with addition denoted $\otimes$ is usually called (max,+) algebra and is an example of dioïd.

We have: $\Re_{\max} = (\Re \cup \{-\infty\}, \oplus, \otimes\}$ with
$a \oplus b = \max(a, b)$; $\varepsilon = -\infty$ is the zero element of $\oplus$
$a \otimes b = a + b$; $e = 0$ is the identity element of $\otimes$
$a \oplus a = a$ (idempotency of $\oplus$)
$a \otimes \varepsilon = \varepsilon \otimes a = \varepsilon$ (absorbing element ε).

The sign $\otimes$ will be omitted as usual when this causes no risk of confusion ($a \otimes b = ab$).

Cyclicity and residuation notions will be used again. Let λ be the maximum mean value of a circuit's weight of a graph associated with a general matrix A. λ is also the maximum eigenvalue of this matrix. A matrix A is cyclic if there exist d and m such that: $(\forall\, i \geq m)\ A^{i+d} = \lambda^d A^i$ with $\lambda^d = d \times \lambda$ in the usual notations.

d is called cyclicity and we say that A is d-cyclic. In this paper, we take the hypothesis that A is d-cyclic.

Theorem 1. Every irreducible matrix is d-cyclic.

The following definition expresses the output trajectory characteristic. It may also be applied to the control or to the desired output after a past evolution of the process.

Definition 2. The output y follows a *d-cyclic* trajectory starting from $k = k_s$ to k_f if $y(k) \geq \lambda^d y(k-d)$ with $k_s \leq k \leq k_f$.

We denote $a \backslash b = \max\{x \mid ax \leq b\}$ the left residuated of b by a (also called the subsolution of equality $ax = b$).

We denote $A \backslash B = \max\{x \mid Ax \leq B\}$ with $A \in \Re_{\max}^{m.n}, x \in \Re_{\max}^{n}, B \in \Re_{\max}^{m}$ and $A \backslash B = A^t \odot B$, where $\odot$ is a matrix product where operation $\oplus$ and $\otimes$ of are replaced respectively by $\wedge$ (minimum) and $\backslash$ of $\Re_{\max}$. The matrix product $\odot$ enables us to calculate easily $A \backslash B$.

3. MODELS

3.1. State equation

In the dioïd (max,+), the model has the following expression

$$\begin{aligned} x(k+1) &= Ax(k) \oplus Bu(k+1) \\ y(k) &= Cx(k) \\ x(0) &= x_0 \end{aligned} \tag{1}$$

where the control u, the output y and the state x are defined on $\Re \cup \{-\infty\}$. In this paper, we consider the Single-Input/Single-Output case. $x(k)$ is a $n.1$ vector of completion times for the k^{th} event.

We note

$$Y_{k_2}^{k_1} = \begin{pmatrix} y(k_1) \\ y(k_1+1) \\ \vdots \\ y(k_2) \end{pmatrix}, \quad U_{k_2}^{k_1} = \begin{pmatrix} u(k_1) \\ u(k_1+1) \\ \vdots \\ u(k_2) \end{pmatrix}$$

and $g_i = CA^iB$. To simplify the notations, we write equally Y for $Y_{k_2}^{k_1}$ and U for $U_{k_2}^{k_1}$, if the context specifies the vectors without ambiguity. From the state equation, we deduce

$$y(k+h) = CA^h x(k) \oplus \sum_{j=0}^{h-1} g_i u(k+h-j)$$

and

$$Y_{k+h}^{k+1} = Hx(k) \oplus GU_{k+h}^{k+1} \quad \text{with} \quad H = \begin{pmatrix} CA \\ CA^2 \\ \vdots \\ CA^{h-2} \\ CA^{h-1} \end{pmatrix}$$

and

$$G = \begin{pmatrix} g_0 & \varepsilon & \dots & \varepsilon & \varepsilon \\ g_1 & g_0 & \dots & \varepsilon & \varepsilon \\ \vdots & \vdots & \dots & \vdots & \vdots \\ g_{h-2} & g_{h-3} & \cdots & g_0 & \varepsilon \\ g_{h-1} & g_{h-2} & \cdots & g_1 & g_0 \end{pmatrix}.$$

3.2. "ARMA" model

Let us recall the principle of the generation of the "ARMA" equations.

From the state equation, we deduce the two following equations:

$$\begin{aligned} \lambda^d y(k-d) &= \lambda^d C A^m x(k-m-d) \oplus \lambda^d \sum_{j=0}^{m-1} g_j u(k-d-j) \\ y(k) &= C A^{m+d} x(k-m-d) \oplus \sum_{j=0}^{m+d-1} g_j u(k-j). \end{aligned}$$

We note

$$a_1 = \sum_{j=0}^{m-1} g_j u(k-d-j); \quad a_2 = \sum_{j=0}^{m+d-1} g_j u(k-j)$$

and we respectively add a_2 and $\lambda^d a_1$ to the previous equations.

$\lambda^d y(k-d) \oplus a_2 = \lambda^d C A^m x(k-m-d) \oplus \lambda^d a_1 \oplus a_2$

$y(k) \oplus \lambda^d a_1 = C A^{m+d} x(k-m-d) \oplus a2 \oplus \lambda^d a_1.$

As the matrix A is cyclic, we can eliminate the state. We deduce

$$y(k) \oplus \lambda^d a_1 = \lambda^d y(k-d) \oplus a_2$$

or

$$y(k) \oplus \lambda^d \sum_{j=0}^{m+d-1} g_{j-d} u(k-j) = \lambda^d y(k-d) \oplus \sum_{j=0}^{m+d-1} g_j u(k-j). \tag{2}$$

Each term contains a single output and a function of the control. However, as the addition does not have the property of symmetry, we cannot express the output $y(k)$ from the other terms. One of the objectives will be to reduce and to exploit this structure.

4. CONTROL SYNTHESIS

4.1. Presentation of the problem

Suppose that some events be designated as controllable, meaning that their input transitions may be delayed from firing until some arbitrary time. The delayed enabling times $u(k)$ for the controllable events are to be provided by a supervisor. Let us suppose that we wish to slow the system down as much as possible without causing any event to occur later than some sequence of execution times Z. We are equally, interested by a regular behavior of the output trajectory and a constraint will be the d-cyclicity: one application is high-frequency transportation systems, for instance [2]. Moreover, we consider a past evolution of the process: it enables changing the desired output, therefore a modification of the production rate. So, let us consider the following problem.

Knowing the dates' values of the control and the output, the number of events being inferior or equal to k_0 and a sequence of the desired output $z(k)$, k ranging from $k_s = k_0 + 1$ to $k_f = k_0 + h$, the problem is to determine the greatest control sequence $u(k)$ such that the output trajectory under the control effect satisfy the following points:

a) each output $y(k)$ occurs at the latest before $z(k)$

b) the output trajectory is d-cyclic.

4.2. Input trajectory

First, we consider the classical problem presented in the introduction. We introduce the following theorem.

Theorem 3. The non-decreasing greatest control such that the output $y(k)$ occur at the latest before the desired output $z(k)$ is given by: for $j = 1$ to h, $u(k_0 + j) = H\backslash Z_{k_0+h}^{k_0+1} = \bigwedge_{i=0}^{h-j} gi\backslash z(k_0 + j + i)$ under the initial constraints $Z_{k_0+h}^{k_0+1} \geq Hx(k_0)$ and $u(k_0 + j) \geq \sum_{i=1}^{n} x_i(k_0); u(k_0 + 1) \geq u(k_0)$.

Proof. We want to calculate the greatest control such that $Y_{k_0+h}^{k_0+1} \leq Z_{k_0+h}^{k_0+1}$. The model is

$Y_{k_0+h}^{k_0+1} = Hx(k_0) \oplus GU_{k_0+h}^{k_0+1}$.

If $Hx(k_0) \not\leq Z_{k_0+h}^{k_0+1}$, the classical problem has no solution.

If $Hx(k_0) \leq Z_{k_0+h}^{k_0+1}$, the greatest control is $H\backslash Z_{k_0+h}^{k_0+1}$. In this case,

$Y_{k_0+h}^{k_0+1} = Hx(k_0) \oplus G(G\backslash Z_{k_0+h}^{k_0+1})$ is maximum and $Y_{k_0+h}^{k_0+1} \leq Z_{k_0+h}^{k_0+1}$. In the single-input single output case, we have $G\backslash Z_{k_0+h}^{k_0+1} = \bigwedge_{i=0}^{h-j} gi\backslash z(k_0 + j + i)$. □

Actually, we can easily prove that this result is another formulation of the Backward equations [1] in a more general case. In the following property, we give another expression of the optimal control for the Backward equations which realizes a connection between the control and the production rate. The calculus is divided into a transient part of length m and a periodic part using the d-cyclicity concept. The following definition expresses the production rate characteristic and can be equally applied to the control or to the desired output after a past evolution of the process.

Proposition 4. Let a desired output trajectory be $(z(k_0+1),\dots,z(k_0+h))^t$. $y(k) = z(k) \wedge \lambda^d \backslash y(k+d)$ with $y(k) = +\infty$ for $k > k_s$. For $j = 1$ to h, $u(k_0+j) = \bigwedge_{i=0}^{m-1} g_i \backslash z(k_0+j+i) \wedge \bigwedge_{i=m}^{m+d-1} g_i \backslash y(k_0+j+i)$ with y d-cyclic.

Proof. If

$$\begin{aligned} Hx(k_0) &\leq Z_{k_0+h}^{k_0+1}, u(k_0+j) = \bigwedge_{i=0}^{h-j} g_i \backslash z(k_0+j+i) \quad \text{(Theorem 2)} \\ u(k_0+j) &= \bigwedge_{i=0}^{m-1} g_i \backslash z(k_0+j+i) \wedge \bigwedge_{i=m}^{h-j} g_i \backslash z(k_0+j+i). \end{aligned}$$

We note

$$u_1(k_0+j) = \bigwedge_{i=0}^{m-1} g_i \backslash z(k_0+j+i)$$

and

$$\begin{aligned} u_2(k_0+j) &= \bigwedge_{i=m}^{h-j} g_i \backslash z(k_0+j+i) \\ u_2(k_0+j) &= \bigwedge_{i=m}^{h-j} g_i \backslash z(k_0+j+i) = \bigwedge_{l=0}^{d-1} \bigwedge_{p=0}^{+\infty} g_{m+l+pd} \backslash z(k_0+j+m+l+pd) \end{aligned}$$

with $z(k_0+j+m+l+pd) = +\infty$ for $j+m+l+pd > h$.

However, $g_{m+l+pd} = (\lambda^d)^p g_{m+l}$ because $g_i = \lambda^d g_{i-d}$ for $i \geq m+d$.

(For example, $g_{m+l+pd} = g_{m+d+l+(p-1)d} = \lambda^d g_{m+l+(p-1)d}$).

So,

$$\begin{aligned} u_2(k_0+j) &= \bigwedge_{l=0}^{d-1} \bigwedge_{p=0}^{+\infty} [(\lambda^d)^p g_{m+l}] \backslash z(k_0+j+m+l+pd) \\ &= \bigwedge_{l=0}^{d-1} g_{m+l} \backslash \left[\bigwedge_{p=0}^{+\infty} (\lambda^d)^p \backslash z(k_0+j+m+l+pd) \right]. \end{aligned}$$

As

$$\begin{aligned} y(k) &= z(k) \bigwedge \lambda^d \backslash y(k+d) = z(k) \bigwedge \lambda^d \backslash [z(k+d) \bigwedge \lambda^d \backslash y(k+2d)] \\ &= z(k) \bigwedge \lambda^d \backslash z(k+d) \bigwedge \lambda^{2d} \backslash y(k+2d) = \dots = \bigwedge_{p=0}^{+\infty} (\lambda^d)^p \backslash z(k+pd), \end{aligned}$$

we finally obtain

$$u_2(k_0+j) = \bigwedge_{l=0}^{d-1} g_{m+l} \backslash y(k_0+j+m+l) = \bigwedge_{i=m}^{m+d-1} g_i \backslash y(k_0+j+i). \qquad \square$$

The following result is immediate.

Proposition 5. Let a d-cyclic desired output trajectory be $(z(k_0+1), \ldots, z(k_0+h))^t$. For $j = 1$ to h, $u(k_0+j) = \bigwedge_{i=0}^{m+d-1} g_i \backslash z(k_0+j+i)$.

Consequently, if the desired trajectory is d-cyclic, the optimal control can be calculated without knowing the values over a horizon of length $d+m$.

Proposition 6. A control sequence deduced from a d-cyclic desired output trajectory z by $u(k_0+j) = \bigwedge_{i=0}^{m+d-1} g_i \backslash z(k_0+j+i)$ is also d-cyclic.

Proof.

$$u(k_0+j) = \bigwedge_{i=0}^{m+d-1} g_i \backslash z(k_0+j+i), \quad u(k_0+j+d) = \bigwedge_{i=0}^{m+d-1} g_i \backslash z(k_0+j+i+d).$$

As $\lambda^d z(k_0+j+i) \leq z(k_0+j+i+d)$, we have

$$g_i \backslash z(k_0+j+i+d) \geq g_i \backslash (\lambda^d z(k_0+j+i)) = \lambda^d [g_i \backslash z(k_0+j+i)].$$

So,

$$\begin{aligned} u(k_0+d+j) &\geq \bigwedge_{i=0}^{m+d-1} \lambda^d [g_i \backslash z(k_0+j+i)] \\ &= \lambda^d \bigwedge_{i=0}^{m+d-1} g_i \backslash z(k_0+j+i) = \lambda^d u(k_0+j). \end{aligned}$$ □

4.3. Output trajectory

In this part, we assume that the control is known. As we have taken the hypothesis that the state is unknown, the problem is to anticipate the effects on the output and to predict it. We shall exploit the "ARMA" structure 2 which is a relation between the input and the output on a finite horizon $m+d$.

In the following theorem, we show that an output trajectory deduced from a d-cyclic input trajectory is also d-cyclic after a transient period and is given by a simple relation.

Theorem 7. If

$$\lambda^d u(k) \leq u(k+d) \quad \text{for} \quad k \geq k_0+1$$

then

$$y(k) = \lambda^d y(k) \oplus \bigwedge_{j=0}^{m+d-1} g_j u(k-j) \quad \text{for} \quad k \geq k_0+m+d.$$

Proof. If u is d-cyclic, then we have $\lambda^d g_{j-d} u(k_0+i-j) \leq g_{j-d} u(k_0+i-j+d)$ for $i \geq j+1$.

If j belongs to $[d, m+d-1]$, then the minimal value of i is $m+d$.

If $i \geq m+d$,

$$\bigwedge_{j=d}^{m+d-1} \lambda^d g_{j-d} u(k_0+i-j) \leq \bigwedge_{j=d}^{m+d-1} g_{j-d} u(k_0+i-j+d)$$
$$= \bigwedge_{j=0}^{m-1} g_j u(k_0+i-j) \leq g_j u(k_0+i-j).$$

As $y(k_0+i) = CA^i x(k_0) \oplus \bigwedge_{j=0}^{i-1} g_j u(k_0+i-j)$, $y(k_0+i) \geq \bigwedge_{j=d}^{m+d-1} \lambda^d g_{j-d} u(k_0+i-j)$.

Finally, we obtain $y(k_0+i) = \lambda^d y(k_0+i-d) \oplus \bigwedge_{j=0}^{m+d-1} g_j u(k_0+i-j)$ for $i \geq m+d$.

□

Theorem 8. If the following initial constraint is verified

$$y(k_0+i) \geq \bigwedge_{j=d\oplus i}^{m+d-1} \lambda^d g_{j-d} u(k_0+i-j) \text{ for } 1 \leq i \leq m+d-1 \tag{3}$$

and if the input is d-cyclic $\lambda^d u(k) \leq u(k+d)$ for $k \geq k_0+1$ then

$$y(k_0+i) = \lambda^d y(k_0+i-d) \oplus \bigwedge_{j=0}^{m+d-1} g_j u(k_0+i-j) \text{ for } i \geq 1. \tag{4}$$

P r o o f. There are three cases:

a) for $i \geq m+d$
We apply the Theorem 7.

b) for $d+1 \leq i \leq m+d-1$
As u is d-cyclic,

$$\sum_{j=d}^{i-1} \lambda^d g_{j-d} u(k_0+i-j) \leq \sum_{j=d}^{i-1} g_{j-d} u(k_0+i-j+d)$$
$$= \sum_{j=0}^{i-d-1} g_j u(k_0+i-j) \leq \sum_{j=0}^{i-1} g_j u(k_0+i-j).$$

As

$$y(k_0+i) = CA^i x(k_0) \oplus \sum_{j=0}^{i-1} g_j u(k_0+i-j),$$

$y(k_0+i)$ is greater than equal $\sum_{j=d}^{i-1} \lambda^d g_{j-d} u(k_0+i-j)$.

As $\sum_{j=d}^{m+d-1} \lambda^d g_{j-d} u(k_0+i-j) = \sum_{j=d}^{i-1} \lambda^d g_{j-d} u(k_0+i-j) \oplus \sum_{j=i}^{m+d-1} \lambda^d g_{j-d} u(k_0+i-j)$,

the condition $y(k_0+i) \geq \sum_{j=d}^{m+d-1} \lambda^d g_{j-d} u(k_0+i-j)$ is reduced to
$y(k_0+i) \geq \sum_{j=i}^{m+d-1} \lambda^d g_{j-d} u(k_0+i-j)$ for $d+1 \leq i \leq m+d-1$.

c) For $1 \leq i \leq d$, the condition remains $y(k_0+i) \geq \sum_{j=d}^{m+d-1} \lambda^d g_{j-d} u(k_0+i-j)$.

We can shortly write $y(k_0+i) \geq \sum_{j=d\oplus i}^{m+d-1} \lambda^d g_{j-d} u(k_0+i-j)$ for $1 \leq i \leq m+d-1$.

Consequently, if u is d-cyclic and the condition 3 holds, the condition $y(k_0+i) \geq \sum_{j=d}^{m+d-1} \lambda^d g_{j-d} u(k_0+i-j)$ is true and we can write the following equality:

$$y(k_0+i) = \lambda^d y(k_0+i-d) \oplus \bigwedge_{j=0}^{m+d-1} g_j u(k_0+i-j) \text{ for } i \geq 1. \qquad \square$$

Remark 1. Let us suppose that the input and output trajectory are known from $k_0 - m - d + 2$ to k_0. We can calculate the right hand term of the inequality 3 from the known values of the control. However, we cannot calculate $y(k_0+i)$ for $1 \leq i \leq m+d-1$ with the equation 4 to verify the condition 3 because this equality needs that condition.

Proposition 9. A sufficient condition of 3 is for $1 \leq i \leq m+d-1$,

$$\sum_{j=0}^{l+i-1} g_j u(k_0+i-j) \geq \sum_{j=d\oplus i}^{m+d-1} \lambda^d g_{j-d} u(k_0+i-j) \text{ with } l = m+d-1. \qquad (5)$$

P r o o f. The state equation gives $y(k_0+i) = CA^{l+i} x(k_0-l) \oplus \sum_{j=0}^{l+i-1} g_j u(k_0+i-j)$.

As $m+d$ is the minimal horizon necessary to exploit the "ARMA" structure, we take $l = m+d-1$ to obtain the maximal information.

So, $y(k_0+i) \geq \sum_{j=0}^{l+i-1} g_j u(k_0+i-j)$ that is the minimal value of the output.

A sufficient condition is for $1 \leq i \leq m+d-1$,

$$\sum_{j=0}^{l+i-1} g_j u(k_0+i-j) \geq \sum_{j=d\oplus i}^{m+d-1} \lambda^d g_{j-d} u(k_0+i-j). \qquad \square$$

4.4. The multi–step control synthesis in the single–input single–output case

The following algorithm gives the solution of the problem of the Section 4.1. when the state is unknown.

a) *d-cyclic desired output trajectory*

We deduce it from

$w(k) = \lambda^d w(k-d) \bigwedge \lambda^d \backslash w(k+d)$ for $k_s \leq k \leq k_f$ with $w(k) = +\infty$ for $k > k_f$.

b) *d-cyclic input trajectory*

The control sequence is deduced by $u(k) = \bigwedge_{i=0}^{m+d-1} g_i \backslash w(k+i)$ for $k_s \leq k \leq k_f$ with the condition $u(k) \geq u(k_0)$.

c) *Predictable output trajectory*

We predict a trajectory y_p with $y_p(k) = \lambda^d y_p(k-d) \oplus \bigwedge_{i=0}^{m+d-1} g_j u(k-i)$ for $k_s \leq k \leq k_f$ with the predictability condition $\bigwedge_{j=0}^{l+i-1} g_j u(k_0+i-j) \geq \bigwedge_{j=d\oplus i}^{m+d-1} \lambda^d g_{j-d} u(k_0+i-j)$ for $1 \leq i \leq m+d-1$ with $l = m+d-1$.

d) *Reachability Analysis*

We verify the following inequality $z(k) \geq y_p(k)$ for $k_s \leq k \leq k_f$.

Proof. The output defined by $w(k) = \lambda^d w(k-d) \bigwedge \lambda^d \backslash w(k+d)$ for $k_s \leq k \leq k_s$ gives the greatest d-cyclic output w such that $\forall k \in [k_s, k_s]$ $w(k) \leq z(k)$ and the Proposition 5 shows that the greatest output control is $u(k) = \bigwedge_{i=0}^{m+d-1} g_i \backslash w(k+i)$. As the desired output trajectory w is d-cyclic, the output control u is also d-cyclic after k_0 (Proposition 6). If the initial constraint 5 is verified and if the input is d-cyclic, then $y_p(k) = \lambda^d y_p(k-d) \oplus \bigwedge_{i=0}^{m+d-1} g_j u(k-i)$ for $i \geq 1$ (Theorem 8 and Proposition 9). Particularly, if the values of $y(k_0+i)$ are known for $1-d \leq i \leq 0$, we can deduce the output trajectory for $1 \leq i \leq k_s - k_0$ that allows us to test the just-in-time criteria. If it exists k such that $z(k) < y_p(k)$, there is not an optimal control such that $z(k) \geq y_p(k)$. □

Remark 2. As the control is applied after the calculus of the control synthesis, the calculated dates must be later than the initial data. Consequently, we have the causality condition $u(k_s) > y(k_0)$.

Remark 3. We can notice that the control is calculated on a finite horizon $d+m$. Precisely, if we consider the case of a d-cyclic desired output trajectory, we have $y(k) = z(k)$ in the first step and the calculus of the control $u(k)$ does not need the values of the desired output over $d+m$. In other words, if the desired output follows the internal rate of the system, the control calculus can consider the real values of the desired output trajectory on only a finite horizon $d+m$ without any optimality reduction: the knowledge of the trajectory under the d-cyclicity hypothesis can be introduced in a sequential and infinite manner. As a result, the desired output trajectory can be easily defined as the infinite repetition of a motif.

4.5. Related work

The reachability analysis verifies the existence of a control that satisfies the constraint a) of the problem and consequently uses only the output trajectory. It is analogous to the existence of non-negative difference between the co-state and the state for the backward approach [1]. In the spirit of the classical automatic control, [12] and [14] consider a strict definition of reachability where the state must exactly be reached. The reachability analysis corresponds partially to the concept of controllable desired output defined in [4, 5] and [6] with a different model. In this work, the state is known and the matrix C equals the identity matrix. The events of the transitions can be delayed or not. In the first case the events are designated as controllable and the matrix B equals I_c: I_c denote the matrix having the identity function on diagonal elements for which the events can be delayed and ε elsewhere. The transposition of the controllable output is $x = A^*(Bu \oplus v) \leq z$ where x, u and z are sequences of firing time vectors for events. v is a sequence of earliest allowable firing time vectors and generalizes the initial condition x_0. To compute the effect of uncontrollable events, the authors choose the equality between the control and the desired output which is a particular choice. The objective of our problem is precisely to determine this control.

In this paper, a basic assumption that allows us to model the system, is that places are First In First Out (FIFO) channels. A place is FIFO if the K^{th} token to enter this place is also the K^{th} that becomes available in this place. The interpretation is that tokens cannot overtake one another which is a necessary numbering condition of the events. We are in this case if the holding times are constant. However, if the holding times vary and if the event numbering is kept, the ARMA equation of the normal system can be used after a delay of $d+m$ occurrences if the system is restored in its usual behavior. This delay is a consequence of the state equation iteration on this horizon [10]. For example, if a place belongs to a cycle that contains one token, the overtaking is forbidden because the place contains at most one token by reason of the property of conservativeness of the event graphs. Consequently, the variation of the holding time does not disrupt the numbering of the state equation. Note that it corresponds to a classical situation where a machine can works on one piece at once. The generalization to the event-index varying case needs a more general Theorem 1 on d-cyclicity. In [4], [5] and [6], the Timed Event Graphs are modeled using a backshift operator which make it possible to consider this case but a difficult problem is to describe algorithms to calculate A star [3].

5. CONCLUSION

In this paper, we present a temporal control synthesis using "ARMA" model in Timed Event Graphs. This approach makes it possible the release of the knowledge of the state vector and enables having a non-stationarity of the model. It enables changes of the desired output and of the production rate in consequence of a modification in the desired output. Coherent with the spirit of the Backwards equations, the solution is modular and can easily be applied.

The control synthesis is based on the d-cyclicity of the desired trajectories relevant to the periodicity of the system and therefore to the production rate. Under this constraint, the desired output trajectory can be defined as the infinite repetition of a motif. In this paper, we also study the "predictability" concept through the "ARMA" model which depends on the system and its behavior. This notion brings up the problem of Observability and Commandability and for the time being, the study of these concepts is an open field for Timed Petri Nets.

APPENDIX

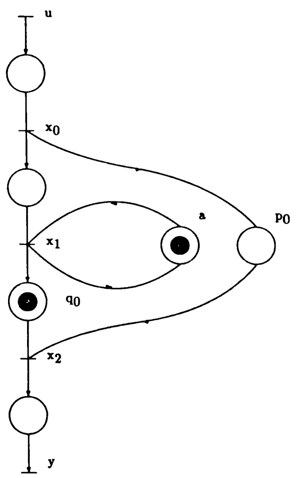

Fig. 1. Timed Event Graph.

A. State equation

We have $y(k) = x_2(k)$; $x_2(k) = q_0 x_1(k-1) \oplus p_0 u(k)$; $x_1(k) = a x_1(k-1) \oplus u(k)$

$$\begin{pmatrix} x_1(k) \\ x_2(k) \end{pmatrix} = \begin{pmatrix} a & \varepsilon \\ q_0 & \varepsilon \end{pmatrix} \begin{pmatrix} x_1(k-1) \\ x_2(k-1) \end{pmatrix} \oplus \begin{pmatrix} e \\ p_0 \end{pmatrix} u(k)$$

$$y(k) = \begin{pmatrix} \varepsilon & e \end{pmatrix} \begin{pmatrix} x_1(k) \\ x_2(k) \end{pmatrix} \quad \begin{pmatrix} x_1(0) \\ x_2(0) \end{pmatrix} = \begin{pmatrix} \varepsilon \\ \varepsilon \end{pmatrix}.$$

B. "ARMA" equation

$$(\forall i \geq 1)\ A^{i+1} = \lambda^d A^i, \quad m = 1, \quad d = 1$$
$$g_0 = CB = p_0, \quad g_1 = CAB = q_0$$
$$CA = (q_0, \varepsilon), \quad CA^2 = (q_0 a, \varepsilon)$$
$$\begin{pmatrix} y(k) \\ y(k+1) \end{pmatrix} = \begin{pmatrix} q_0 & \varepsilon \\ q_0 a & \varepsilon \end{pmatrix} x(k-1) \oplus \begin{pmatrix} p_0 & \varepsilon \\ q_0 & p_0 \end{pmatrix} \begin{pmatrix} u(k) \\ u(k+1) \end{pmatrix}$$
$$y(k+1) \oplus a g_0 u(k) = a y(k) \oplus g_1 u(k) \oplus g_0 u(k+1).$$

C. Control synthesis

a) $w(k) = z(k) \bigwedge a \backslash w(k+1)$

b) $u(k) = g_0 \backslash w(k) \bigwedge g_1 \backslash w(k+1)$

c) $y_p(k+1) = a y_p(k) \oplus g_1 u(k) \oplus g_0 u(k+1).$

Predictability condition: $g_0 u(k_0+1) \oplus g_1 u(k_0) \geq a g_0 u(k_0)$.
Causality condition: $u(k_s) > y(k_0)$.

Let $p_0 = 3$, $q_0 = 1$, $a = 2$.

If the holding times are constant, we can have for example the following evolution·

k	0	1	2	3
u		e	1	3
x_1	ε	e	2	4
x_2	ε	3	4	6
y	ε	3	4	4

But we assume that the holding time "a" of the recycled place has undergone a variation at $k = 2$.

We have $x_1(k) = a x_1(k-1) \oplus u(k)$ with $a = 4$ for $k = 2$ and $a = 2$ otherwise.

k	0	1	2	3
u		e	1	3
x_1	ε	e	4	6
x_2	ε	3	4	6
y	ε	3	4	6

Let a desired output trajectory be $(z(4),\ z(5),\ z(6))^t = (11, 14, 14)^t \cdot (k_0 = 3,\ k_s = 4,\ k_f = 6)$. We deduce a d-cyclic trajectory $(w(4),\ w(5),\ w(6))^t = (10, 12, 14)^t$, hence the control $(u(4),\ u(5),\ u(6))^t = (7, 9, 11)^t$.

The predictable output trajectory y_p is: $(y_p(4),\ y_p(5),\ y_p(6))^t = (10, 12, 14)^t$. The predictability condition $g_0 u(k_0+1) \oplus g_1 u(k_0) \geq a g_0 u(k_0)$ is verified $(u(k_0) = 3,\ u(k_0+1) = 7)$ as the causality condition $u(k_s) = 7 > y\ (k_0) = 6$.

Using the state equations, the simulation confirms this result.

k	0	1	2	3	4	5	6
u		e	1	3	7	9	11
x_1	ε	e	4	6	8	10	12
x_2	ε	3	4	6	10	12	14
y	ε	3	4	6	10	12	14

(Received April 8, 1998.)

REFERENCES

[1] F. Baccelli, G. Cohen, G. J. Olsder and J. P. Quadrat: Synchronization and Linearity. An Algebra for Discrete Event Systems. Wiley, New York 1992.

[2] H. Bracker: Algorithms and Applications in Timed Discrete Event Systems. Ph.D thesis, Delft University of Technology, 1993.

[3] G. P. Brat and V. K. Garg: A max-plus algebra of signals for the supervisory control of real-time discrete event systems. In: 9th Symposium of the IFAC on Information Control in Manufacturing, Nancy-Metz 1998.

[4] D. D. Cofer and V. K. Garg: A timed model for the control of discrete event systems involving decisions in the max/plus algebra. In: Proc. 31st Conference on Decision and Control, Tucson 1992.

[5] D. D. Cofer: Control and Analysis of Real-Time Discrete Event Systems. Ph.D. Thesis, University of Texas, Austin 1995.

[6] D. D. Cofer and V. K. Garg: Supervisory control of real-time discrete-event systems using lattice theory. IEEE Trans. Automat. Control *41* (1996), 2, 199–209.

[7] G. Cohen, S. Gaubert and J.-P. Quadrat: From first to second-order theory of linear discrete event systems. In: 1st IFAC World Congress, Sydney 1993.

[8] Ph. Declerck: "ARMA" model and admissible trajectories in timed event graphs. In: CESA'96, IMACS, IEEE-SMC, Lille 1996.

[9] Ph. Declerck: Control synthesis using the state equations and the "ARMA" model in timed event graphs. In: 5th IEEE Mediterranean Conference on Control and Systems, Paphos 1997.

[10] Ph. Declerck and M. Mares: Temporal control synthesis and failure recovery. In: 9th Symposium of the IFAC on Information Control in Manufacturing, Nancy-Metz 1998.

[11] S. Gaubert: Théorie des systèmes linéaires dans les dioïdes. Ph.D Thesis, Ecole des Mines de Paris 1992.

[12] M. J. Gazarik and E. W. Kamen: Reachability and observability of linear systems over max-plus. In: 5th IEEE Mediterranean Conference on Control and Systems, Paphos 1997, revised version: Kybernetika *35* (1999), 2–12.

[13] M. Gondran and M. Minoux: Graphes et algorithmes. Edition Eyrolles 1995.

[14] J.-M. Prou and E. Wagneur: Controllability in the max-algebra. In: 5th IEEE Mediterranean Conference on Control and Systems, Paphos 1997, revised version: Kybernetika *35* (1999), 13–24.

Philippe Declerck, Maître de Conférences at the University of Angers, LAUPERA-ISTIA, 62 avenue Notre-Dame du Lac, F-49000 Angers. France.
e-mail: philippe.declerck@istia.univ-angers.fr

Ph. Declerck et M.K. Didi Alaoui,

Optimal control synthesis of timed event graphs with interval model specifications, IEEE Transactions on Automatic Control, vol.55, n°2, IETAA9 , pages 518-523, février 2010.

Optimal control synthesis of timed event graphs with interval model specifications

Philippe Declerck and Mohamed Khalid Didi Alaoui

***Abstract*—The purpose of this paper is the optimal control synthesis of a Timed Event Graph when the state and control trajectories should follow the specifications defined by an interval model. The problem is reformulated in the fixed point form and the spectral theory gives the conditions of existence of a solution.**

***Index Terms*—Timed Event Graphs, P-time Petri nets, Time Stream Petri nets, (min, max, +) functions, cycle-time vector, fixed point.**

I. Introduction

In [7] and [6], we have shown that P-time Event Graphs and Time Stream Event Graphs can be modeled by a new class of systems called interval systems, for which the time evolution is not strictly deterministic, but is described by intervals which use the operations of maximization, minimization and addition to define the lower and upper bound constraints. The consistency of P-time Event Graphs can also be studied in tropical algebra using the spectral vector [6]. In this paper, we focus on the following problem.

Some events are stated as controllable, meaning that the firing of the corresponding transitions (input) may be delayed until some arbitrary time provided by a supervisor. Considering a desired behavior of some transitions (output) of the Timed Event Graph such as a sequence of execution times (desired output), we wish to slow down the system as much as possible ensuring all output events to occur before their desired execution time.

Moreover, we assume that the Timed Event Graph must follow specifications defined by a second model. Expressed by an interval model, this desired behavior is expressed by inequalities which introduce lower and upper bounds on the dates of firing of all the transitions. As the specifications can be incoherent or too restrictive with respect to the Timed Event Graph, a first problem is to determine whether there exist control actions which will restrict the system to that behavior. If the trajectories of the Timed Event Graph can follow additional specifications, a second objective is to determine the greatest input in order to obtain the desired behavior defined by the static constraints (the desired output) and the dynamic constraints (expressed by an interval model).

In this field, a first class of approaches analyzes the state space and develops controllers in order to keep trajectories inside a space deduced from a given specification. The aim is the extension of the concept of (A,B)-invariant subspace to linear systems over the max-plus semiring. The computation of the maximal set of the initial states is analyzed in [2] [11]. Another group of methods [13] [7] [6] considers extremal points of the state space and develops optimal control in order to keep trajectories close to a reference trajectory following additional constraints. The aim is the extension of the principle of the well-known model predictive control.

Let us point out two main differences between these two classes of methods. In feedback approaches, the technique is based on the addition of new structures such as loops which modifies the initial Timed Event Graph. This technique can improve the boundedness of the Petri net but also reduce its production rate and modify its liveness. In predictive approaches, the Petri Net and all its characteristics are kept as we can assume that a preparation composed of scheduling, resources optimisation,... has established an optimized model. Another difference is that predictive approaches can be applied to a large class of models. Particularly, approaches based on a feedback defined by a Petri net are limited by the condition that the temporisation and initial marking of each added place are non-negative. The existence of a linear state feedback is discussed in [11] (see part V and in particular example 4): this problem is reminiscent of difficulties of the theory of linear dynamical systems over rings [10].

In [7] and [6], the control synthesis has been considered by the authors for P-time Event Graphs and Time Stream Event Graphs. Extending these studies, the present paper proposes the control synthesis of a Timed Event Graph (plant) following an interval system (specifications) which can express a set of Event Graphs. Based on a fixed point technique, the main advantage of the approach is that it allows adding (min, max, +) lower and upper bound constraints on all the transitions. It contains new material including a new fixed-point algorithm and an everyday example.

In this paper, the usual hypothesis that places of the Event Graphs should be First In First Out (FIFO) is taken. No hypothesis is taken on the structure of the Event Graphs, which does not need to be strongly connected. No closed-loop structure of control is given a priori. The initial marking should only satisfy the classical liveness condition.

In the next part, we present the optimal control problem and we analyze the existence of a finite solution through spectral theory. This section is followed by an algorithm which determines the greatest control. The notations and a brief review of preliminary results are presented in the Appendix.

P. Declerck and M.K Didi Alaoui are with LISA EA4014, University of Angers, 62 avenue Notre-Dame du Lac, 49000 Angers, France (tel: +33(0)2.41.22.65.60; fax: +33(0)2.41.22.65.61; e-mail: philippe.declerck@univ-angers.fr; mohamed.didi.alaoui@univ-angers.fr)

II. Control problem

By usual (max,+) algebraic notation, maximization, minimization and addition operations are noted respectively $\oplus$, $\wedge$ and $\otimes$. Variable $x_i(k)$ (respectively, $u_i(k)$) is the date of the k^{th} firing of internal transition x_i (respectively, of input transition u_i). Assuming that the following models are available for $k \geq k_s + 1$ with $k \in \mathbb{Z}$ ('s' for 'start'), we consider the control of an event graph modeled as a (max, +) system

$$x(k) = A_1 \otimes x(k-1) \oplus A_0 \otimes x(k) \oplus B_0 \otimes u(k) \quad , \quad (1)$$

with state x subject to the following interval constraints

$$f^-(x(k-1), x(k), u(k)) \leq x(k) \leq f^+(x(k-1), x(k), u(k)) \ , \quad (2)$$

and the desired output $y(k) \leq z(k)$ where

$$y(k) = C \otimes x(k) \ ,$$

is the output of the event graph and $z(k)$ for $k \in [k_s + 1, k_f]$ ('f' for 'final') is the desired output. The goal is to find the greatest control $u(k)$ for $k \in [k_s + 1, k_f]$.

In this paper, the following usual assumptions on the plant defined by Timed Event Graph (1) are made. The Timed Event Graph is structurally observable and controllable [1]: every internal transition can be reached by a path from one input transition at least and, from every internal transition, there exists a path to one output transition at least. The structural observability (respectively controllability) gives a condition to observe an effect in the output (resp. transition) whose origin comes from one internal transition (resp. input) at least. We also suppose that the system works at the earliest time and the Timed Event Graph is defined by equality (1).

Moreover, the following assumption on additional specifications defined by interval model (2) is made: the lower bound f^- is max-only (see appendix) and is defined by $f^-(x(k-1), x(k), u(k)) = A_1^- \otimes x(k-1) \oplus A_0^- \otimes x(k) \oplus B^- \otimes u(k)$. No assumption is made on f^+ which is a (min, max, +) function. Except the form of functions in (2), there is no assumption on the interval model which can describe a set of non-connected Event Graphs.

Example. A simple real-world problem: education system

Plant. Described by the Timed Event Graph in Figure 1, the plant corresponds to the following work of a professor: the lesson is composed of a lecture (duration: T_1) followed by practical work (duration: T_2). Transitions express the following events. x_1: beginning of the lecture; x_2: beginning of practical work; x_3: end of practical work; u_1: decision to start the lecture; u_2: decision to start practical work. From Figure 1, we can deduce the following matrices.

$$A_0 = \begin{pmatrix} -\infty & -\infty & -\infty \\ T_1 & -\infty & -\infty \\ -\infty & T_2 & -\infty \end{pmatrix} , \ B_0 = \begin{pmatrix} 0 & -\infty \\ -\infty & 0 \\ -\infty & -\infty \end{pmatrix}$$

and $C = \begin{pmatrix} -\infty & -\infty & 0 \end{pmatrix}$. As the state trajectory is non-decreasing ($x(k) \geq x(k-1)$), $A_1 = I$.

Interval system. Moreover, the teacher must follow the official instructions: a lesson must not exceed T_3^+ and not be less than T_3^-. These specifications can be described by a

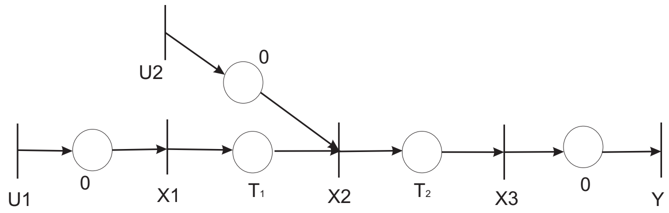

Fig. 1. Timed Event Graph (plant: tasks of the professor)

new Event Graph which can be a simple P-time Event Graph. As $T_3^- \otimes x_1(k) \leq x_3(k) \leq T_3^+ \otimes x_1(k)$, the corresponding interval system is as follows.

$$A_1^- = -\infty, \ A_0^- = \begin{pmatrix} -\infty & -\infty & -\infty \\ -\infty & -\infty & -\infty \\ T_3^- & -\infty & -\infty \end{pmatrix}, \ B^- = -\infty$$

and $f^+(x(k-1), x(k), u(k)) = \begin{pmatrix} +\infty \\ +\infty \\ T_3^+ \otimes x_1(k) \end{pmatrix}$.

Problem. The lesson must stop before the daily closing time of school which corresponds to the desired output z. A problem can be the determination of the latest times to begin the lesson such that each specification is satisfied. If the teacher begins the lesson after this date which corresponds to the greatest control u, the practical work cannot be finished ($y \not\leq z$).

III. Approach

In the sequel, the problem is reformulated as a fixed point problem which describes all the relations between components of state trajectory and control trajectory. A relaxation presented in part III-B allows the analysis of existence of a finite control while the determination of the greatest control $u(k)$ for $k \in [k_s+1, k_f]$ is made in part III-C. The resolution also proposes a state trajectory $x(k)$ for $k \in [k_s, k_f]$ which can be followed by the system if the corresponding control $u(k)$ is applied.

A. *Fixed point form*

Any feasible control trajectory $u(k)$ to the control problem is considered in the following theorem.

Theorem 1: Given the desired output $z(k)$, the system composed of models (1), (2) and constraint $y(k) = C \otimes x(k) \leq z(k)$ where $k \in [k_s + 1, k_f]$, is equivalent to the following inequality system:

$$\left\{ \begin{array}{l} x(k) \leq C \backslash z(k) \\ \wedge [A_0 \oplus A_0^-] \backslash x(k) \wedge g^+(x(k-1), x(k), u(k)) \\ x(k-1) \leq [A_1 \oplus A_1^-] \backslash x(k) \\ u(k) \leq [B_0 \oplus B^-] \backslash x(k) \end{array} \right. , \quad (3)$$

where $k \in [k_s + 1, k_f]$ and, $g^+(x(k-1), x(k), u(k)) =$

$$[A_1 \otimes x(k-1) \oplus A_0 \otimes x(k) \oplus B_0 \otimes u(k)] \wedge f^+(x(k-1), x(k), u(k)) \ . \quad (4)$$

Proof:
Equality (1) is equivalent to

$$\left\{ \begin{array}{l} A_1 \otimes x(k-1) \oplus A_0 \otimes x(k) \oplus B_0 \otimes u(k) \leq x(k) \\ x(k) \leq A_1 \otimes x(k-1) \oplus A_0 \otimes x(k) \oplus B_0 \otimes u(k) \end{array} \right. , \quad (5)$$

and using the residuation (see Appendix), we obtain

$$\left\{\begin{array}{l} x(k-1) \leq A_1 \backslash x(k) \\ x(k) \leq A_0 \backslash x(k) \wedge [A_1 \otimes x(k-1) \oplus A_0 \otimes x(k) \\ \oplus B_0 \otimes u(k)] \\ u(k) \leq B_0 \backslash x(k) \end{array}\right. . \quad (6)$$

From (2), we similarly obtain

$$\left\{\begin{array}{l} x(k-1) \leq A_1^- \backslash x(k) \\ x(k) \leq A_0^- \backslash x(k) \wedge f^+(x(k-1), x(k), u(k)) \\ u(k) \leq B^- \backslash x(k) \end{array}\right. . \quad (7)$$

Finally, system (3) is found after adding the classical relation $x(k) \leq C \backslash z(k)$ which is directly deduced from $C \otimes x(k) \leq z(k)$. ■

The above equivalences are based on well-known properties of residuation (see Appendix).

Note that we can deduce an upper bound on the state by using $x(k) \leq C \backslash z(k)$ knowing $z(k)$. Using this calculated upper bound, we can calculate an upper bound on the control by using the last inequality of (3). Let us develop system (3) algebraically on horizon $[k_s, k_f]$.

$$\left\{\begin{array}{l} x(k_s) \leq [A_1 \oplus A_1^-] \backslash x(k_s+1) \\ x(k) \leq C \backslash z(k) \wedge [A_0 \oplus A_0^-] \backslash x(k) \wedge [A_1 \oplus A_1^-] \backslash x(k+1) \\ \wedge g^+(x(k-1), x(k), u(k)) \text{ for } k \in [k_s+1, k_f-1] \\ u(k) \leq [B_0 \oplus B^-] \backslash x(k) \text{ for } k \in [k_s+1, k_f-1] \\ x(k_f) \leq C \backslash z(k_f) \wedge [A_0 \oplus A_0^-] \backslash x(k_f) \\ \wedge g^+(x(k_f-1), x(k_f), u(k_f)) \\ u(k_f) \leq [B_0 \oplus B^-] \backslash x(k_f) \end{array}\right. \quad (8)$$

We now introduce the following notation. Denoted X_l, vector $(x(k_s)^t, x(k_s+1)^t, u(k_s+1)^t, x(k_s+2)^t, u(k_s+2)^t, ..., x(k_s+l)^t, u(k_s+l)^t)^t$ is the concatenation of state trajectory $(x(k_s)^t, x(k_s+1)^t, x(k_s+2)^t, ..., x(k_s+l)^t)^t$ and input trajectory $(u(k_s+1)^t, u(k_s+2)^t, ..., u(k_s+l)^t)^t$ where $l = k_f - k_s$ denotes the length of the horizon. System (8) can be rewritten as the following fixed point form

$$X_l \leq h_l(X_l) \quad , \quad (9)$$

where h_l is clearly a (min, max, +) function. Therefore, we must analyze and solve a fixed-point problem of type $x \leq f(x)$ (if x exists) over horizon $[k_s, k_f]$. System (9) contains a "Backward" part as $x(k-1) \leq [A_1 \oplus A_1^-] \backslash x(k)$ but also, a "Forward" part with expression $x(k) \leq g^+(x(k-1), x(k), u(k))$. This fact forbids the immediate writing of forward or backward recurrences such as the state equations or the classical backward equations used in control for Timed Event Graphs. Let us note that g is a (min, max, +) function even if there is no function $f^+(x(k-1), x(k), u(k))$ in the specifications.

B. Existence of a finite solution

The aim of this part is to verify the existence of a finite solution (not just the greatest solution) in the control synthesis. Presented in the appendix, the property of homogeneity of $(\min, \max, +)$ functions belonging to $F(n, n)$ is necessary to use Theorem 3. However, function h_l in (9) is not homogeneous as h_l contains desired output z which is a datum of the problem (see terms $C \backslash z(k)$ in (8)). To apply the spectral theory, we will use a relaxation by associating a new variable x_0 with every constant which leads to slight modifications of (8). All terms of system (8) are kept except terms $C \backslash z(k)$ which become $C \backslash (z(k) \otimes x_0)$ for $k \in [k_s+1, k_f]$. Moreover, inequality $x_0 \leq x_0$ which is always satisfied, is added and consequently, the system defined by (9) can also be defined by following system (10) with condition $x_0 = 0$. Therefore, with nonhomogeneous function h_l is associated a homogeneous function H_l, denoted with the same letters but in capitals.

$$\begin{pmatrix} x_0 \\ X_l \end{pmatrix} \leq H_l \begin{pmatrix} x_0 \\ X_l \end{pmatrix} \quad (10)$$

If $\begin{pmatrix} x_0 \\ X_l \end{pmatrix}$ is an arbitrary solution of (10) in $\mathbb{R}$ (without condition $x_0 = 0$), then

$$(x_0)^{-1} \otimes \begin{pmatrix} x_0 \\ X_l \end{pmatrix} = (-x_0) \otimes \begin{pmatrix} x_0 \\ X_l \end{pmatrix} = \begin{pmatrix} 0 \\ (-x_0) \otimes X_l \end{pmatrix} , \quad (11)$$

is a solution of (10) with condition $x_0 = 0$. We can interpret variable x_0 as a possibly negative period added to the desired output which delays or anticipates all calculated dates with respect to the origin of time. Particularly, all the input dates can be postponed or anticipated with the same duration. A similar technique is used in part V of [11] and the relevant variable can be interpreted as an increase or decrease of every temporisation of the Timed Event Graph.

Using the cycle time vector χ (see Appendix), the following theorem analyzes the existence of a finite vector X_l. Therefore, it gives the conditions such that the plant can follow a finite state trajectory obeying the specifications.

Theorem 2: There exists a finite vector X_l satisfying models (1), (2) and constraint $y(k) = C \otimes x(k) \leq z(k)$ on horizon $[k_s, k_f]$ if and only if $\chi(H_l) \geq 0$.

Proof: As H_l is a homogeneous function, spectral vector $\chi(H_l)$ can be calculated and Theorem 3 (see appendix) applies. If the cycle time satisfies the corresponding condition of existence, system (10) has a solution $\begin{pmatrix} x_0 \\ X_l \end{pmatrix}$. For any solution, an obvious translation can be applied in such a way that the first component x_0 equals zero. If $x_0 = 0$, then this solution satisfies (8) which is equivalent to system which is made up of (1), (2) and constraint $y(k) = C \otimes x(k) \leq z(k)$. ■

Example (education system continued)

The calculation of the spectral vector of the above function denoted H_1 leads to $\chi(H_1) = 0$ for the following values: $T_1 = 60; T_2 = 90; T_3^- = 165; T_3^+ = 175$. Therefore, the above system is consistent. The following subsystem can be deduced from (8).

$$\left\{\begin{array}{l} x_1(1) \leq -T_1 + x_2(1) \\ x_2(1) \leq -T_2 + x_3(1) \\ x_3(1) \leq +T_3^+ + x_1(1) \end{array}\right. \quad (12)$$

Therefore, $x_1(1) \leq -T_1 - T_2 + T_3^+ + x_1(1)$. If $T_1 = 60$, $T_2 = 90$ and $T_3^+ = 175$, the inequality becomes $x_1(1) \leq -60 - 90 + 175 + x_1(1) = 25 + x_1(1)$ which is consistent.

Now, if we take $T_3^- = 135$ and $T_3^+ = 145$, we obtain $-T_1 - T_2 + T_3^+ = -5 < 0$. An incoherency appears as inequality $x_1(1) \leq -T_1 - T_2 + T_3^+ + x_1(1)$ gives $x_1(1) \leq -5 + x_1(1)$: the interpretation is that the lesson time of the professor is inconsistent with the official instructions. The calculation shows that several components of spectral vector $\chi(H_1)$ are negative: $\chi(H_1) = (0, -\frac{5}{3}, -\frac{5}{3}, ..., -\frac{5}{3})^t$. Consequently, $\chi(H_1) \ngeq 0$ and the above system (10) has no solution. ■

C. Determination of the greatest solution

The previous part considers the existence of an arbitrary finite solution X_l. Let us now consider a particular solution which is the greatest solution. The greatest control trajectory and also, the greatest state trajectory are clearly found if the greatest solution is determined.

The existence of the greatest solution on complete lattices can be proved by using the famous fixed point theorem of Knaster-Tarski [14] whose conditions are already satisfied: in our problem, $h_l(.)$ is an isotone function defined on a complete lattice $\overline{\mathbb{R}}_{max}$=($\mathbb{R} \cup \{-\infty\} \cup \{+\infty\}, \leq$). If an algorithm gives the greatest solution of $x \leq f(x)$, this solution also satisfies the relevant equality. Function $h_l(.)$ is also discontinuous but Knaster-Tarski Theorem does not require continuity of the function.

The effective calculation of the greatest control of inequality (9) can be made by a classical iterative algorithm of Mc Millan and Dill [12] which particularizes the algorithm of Kleene to (min, max, +) expressions. The general resolution of $x \leq f(x)$ is given by the iterations of $x_i \leftarrow x_{i-1} \wedge f(x_{i-1})$ if the finite starting point is greater than the final solution. Here, number i represents the number of iterations and not the number of components of vector x. Following this framework, we provide an algorithm specific to the determination of the greatest state and control. Described below, algorithm 1 uses a decomposition of system (8) in its backward part and forward part. For instance, the second relation of (8) is divided into two parts

$$\begin{cases} x(k) \leq C\backslash z(k) \wedge [A_0 \oplus A_0^-]\backslash x(k) \wedge [A_1 \oplus A_1^-]\backslash x(k+1) \\ x(k) \leq g^+(x(k-1), x(k), u(k)) \end{cases} \quad (13)$$

for $k \in [k_s + 1, k_f - 1]$. In the following algorithm, the input is the desired output trajectory $z(k)$ from $k = k_s + 1$ to k_f. The outputs are the greatest control trajectory $u(k)$ from $k = k_s + 1$ to k_f and the greatest state trajectory $x(k)$ from $k = k_s$ to k_f . Term $x(k, i)$ is the state estimate at event number k and iteration i.

Algorithm 1

Step 0 (initialization): $i = 1$; $x(k, i-1) \leftarrow T$ for $k \in [k_s, k_f]$

Repeat

- Step 1: backward calculation of the state

$x(k_f, i) \leftarrow x(k_f, i-1) \wedge C\backslash z(k_f) \wedge [A_0 \oplus A_0^-]\backslash x(k_f, i-1)$

$x(k, i) \leftarrow x(k, i-1) \wedge C\backslash z(k) \wedge [A_0 \oplus A_0^-]\backslash x(k, i-1) \wedge [A_1 \oplus A_1^-]\backslash x(k+1, i)$ from $k = k_f - 1$ to $k_s + 1$

$x(k_s, i) \leftarrow x(k_s, i-1) \wedge [A_1 \oplus A_1^-]\backslash x(k_s + 1, i)$

- Step 2: backward calculation of the control

$u(k) \leftarrow (B_0 \oplus B^-)\backslash x(k, i)$ from $k = k_f$ to $k_s + 1$

- Step 3: forward calculation of the state

$x(k, i) \leftarrow x(k, i) \wedge g^+(x(k-1, i), x(k, i), u(k))$ from $k = k_s + 1$ to k_f

Until no $x(k, i)$ changes for $k_s \leq k \leq k_f$

Using a "Backward" approach, the first iteration of step 1 allows the determination of the starting state trajectory of the general algorithm. In step 2, the control is directly calculated by a unique relation from the state and the memorization of their previous calculated values is useless as $x(k, i)$ is minimized at each step i. In step 3, state minimization improves the calculated values of step 1 by a forward recurrence. These steps are repeated until convergence. When the minimization of the state stops, the algorithm gives the optimal state and control which satisfy the inequalities of the plant (described by a Timed Event Graph (1)) following the specifications (expressed by an interval system (2)).

Step 1 corresponds to the well-known backward equality described in part 5.6 in [1] if we consider the case of a Timed Event Graph without specification. Classical handlings can reduce the writing of step 1 and a relation similar to (5.62) in [1] can be obtained. Let us recall that the greatest solution (the latest times) of the control problem is explicitly given by the "Backward" recursive equations. The development of the algorithm is easy and only requires the memorization of the matrices of the different models and the estimated trajectory $x(k, i)$. In the general case, it is difficult to carry out a theoretical analysis of the number of iterations as in many algorithms in this field [3]. The general algorithm of Mc Millan and Dill [12] is known to be pseudo-polynomial in practice.

Example (education system continued)

The lesson must now stop before the daily closing time of school from Monday (first day: $k = 1$) to Friday ($k = 5$): Assume that desired output sequence $z(k)$ from $k = 1$ to 5 is 1140, 2580, 3600, 5460, 6480. So, $k_s = 0$ and $k_f = 5$. The corresponding output sequence is as follows.

,

k	1	2	3	4	5
u_1	975	2415	3435	5295	6315
u_2	1050	2490	3510	5370	6390

■

IV. Conclusion

This paper solves the problem of optimal control synthesis of a Timed Event Graph when the state and control trajectories are constrained by specifications defined by an interval model. The interval descriptor system can describe the time behavior of a lot of models such as Timed Event Graphs, P-time Petri nets and Time Stream Event Graphs for semantic rules "And" and "Weak-And" [4]. The problem is reformulated in a fixed point form. The spectral theory gives conditions of existence of a solution while a proposed algorithm makes it possible to determine the greatest state and control. The application of the calculated control generates a state trajectory obeying the specifications.

V. Appendix

In this section, we shall review a few basic theoretical notions about dioids. For more extensive presentations, the reader is invited to consult the following references: [1] and [5].

A monoid is a couple $(S, \oplus)$ where operation $\oplus$ is associative and presents a neutral element. A semi-ring S is a triplet $(S, \oplus, \otimes)$ where $(S, \oplus)$ and $(S, \otimes)$ are monoids, $\oplus$ is commutative, $\otimes$ is distributive in relation to $\oplus$ and the zero element ε of $\oplus$ is the absorbing element of $\otimes$ ($\varepsilon \otimes a = a \otimes \varepsilon = \varepsilon$). A dioid D is an idempotent semi-ring (operation $\oplus$ is idempotent, that is $a \oplus a = a$). Let us note that unlike the structures of group and ring, monoids and semi-rings do not have a property of symmetry on S. The set $\mathbb{R} \cup \{-\infty\}$ provided with the maximum operation denoted $\oplus$ and the addition denoted $\otimes$ is an example of dioid which is usually denoted $\mathbb{R}_{max}$ = $(\mathbb{R} \cup \{-\infty\}, \oplus, \otimes)$. The neutral elements of $\oplus$ and $\otimes$ are represented by $\varepsilon = -\infty$ and $e = 0$, respectively.

A dioid D is complete if it is closed for infinite sums and the distributivity of the multiplication with respect to addition extends to infinite sums: $(\forall\ c \in D\ )\ (\forall\ A \subseteq D)\ c \otimes (\bigoplus_{x \in A} x) = \bigoplus_{x \in A} c \otimes x$. For example, $(\mathbb{R} \cup \{-\infty\} \cup \{+\infty\}, \oplus, \otimes)$ usually denoted $\overline{\mathbb{R}}_{max}$, is complete. The set of nxn matrices with entries in a complete dioid D included with the two operations $\oplus$ and $\otimes$ is also a complete dioid, which is denoted $D^{n\text{x}n}$. Nonsquare matrices can be considered if they are completed with rows or columns with entries equal to ε. The sum and product of matrices are defined conventionally from the sum and product in D.

Let Γ be a subset of vectors over $\overline{\mathbb{R}}_{max}$. The partial order denoted $\leq$ is defined as follows: $v \leq w \Longleftrightarrow v \oplus w = w$. It is also a componentwise order which allows the comparison of any pair of vectors (v, w) i.e. $v \leq w \Longleftrightarrow v_i \leq w_i$, for each component i. In the paper, this concept is applied to control and state trajectories. The element v of subset Γ is called greatest element or maximum element if and only if $w \leq v$ for all $w \in \Gamma$. In other words, it is greater than any other element of the subset: (see part 4.3.1 of [1] for more details). If this greatest element exists, it is unique as the existence of two different maximum elements v and w implies $w \leq v$ and $v \leq w$. Let $v \wedge w$ denote the lower bound of v and w.

A mapping f is monotone or isotone if $x \leq y$ implies $f(x) \leq f(y)$. Let f: $E \rightarrow F$ be an isotone mapping, where (E, $\leq$) and (F,$\leq$) are ordered sets. Mapping f is said to be residuated if for all $y \in F$, the least upper bound of subset $\{x \in E \mid f(x) \leq y\}$ exists and belongs to this subset. The corresponding mapping, denoted $f^d(y)$ is called the residual of f. When f is residuated, f^d is the only isotone mapping, such that $f \circ f^d \leq Id_F$ and $f^d \circ f \geq Id_E$ where Id_F and Id_E are identity mappings. Mapping $x \in (\overline{\mathbb{R}}_{max})^n \mapsto A \otimes x$, defined over $\overline{\mathbb{R}}_{max}$ is residuated [1] and the left $\otimes-$ residual of b by A is denoted by: $A \backslash b = \max\{x \in (\overline{\mathbb{R}}_{max})^n$ such that $A \otimes x \leq b\}$. Moreover, $(A \backslash b)_i = \bigwedge_{j=1}^{m} A_{ji} \backslash b_j$ where A is an $m \times n$ matrix.

A $(\min, \max, +)$ function of set $F(n, 1)$ is any function $f : \mathbb{R}^n \longrightarrow \mathbb{R}^1$, which can be written as a term in the following grammar: $f = x_1, x_2, \ldots, x_n \mid f \otimes a \mid f \wedge f \mid f \oplus f$ where a is an arbitrary real number ($a \in \mathbb{R}$). The vertical bars separate the different ways in which terms can be recursively constructed. A $(\min, \max, +)$ function of set $F(n, m)$ is any function f : $\mathbb{R}^n \longrightarrow \mathbb{R}^m$, such that each component f_i is a $(\min, \max,$ +) function of $F(n, 1)$.

Let $f \in F(n, 1)$. If f can be represented by a term that does not use $\wedge$, it is said to be max-only or (max, +). If f can be represented by a term that does not use $\oplus$, it is said to be min-only or (min, +). If f can be represented by a term that does not use $\wedge$ and $\oplus$, it is said to be simple. As the type of the interval model is defined by the types of functions f^- and $f^+ \in F(n, m)$, we can characterize the model by the following pair (type of f^-, type of f^+) which defines different types of systems. Type ((min, max, +), (min, max, +)) naturally represents the more general mathematical case.

The following iterative form where number i represents the number of iterations, is considered: $x(i) = f(x(i-1)), \forall i \geq 1$ and $x(0) = \xi \in \mathbb{R}^n$, where f is a $(\min, \max, +)$ function of $F(n, n)$. Every function of $F(n, n)$ has the property of homogeneity which is defined as follows: $\forall \lambda \in \mathbb{R}, \forall x \in \mathbb{R}^n$ $f(\lambda \otimes x) = \lambda \otimes f(x)$ in the usual vector-scalar convention: $(\lambda \otimes x)_i = \lambda \otimes x_i$.

In the following fundamental theorem, the notion of cycle time makes it possible to verify the existence of a solution of different inequalities and equalities. The cycle time vector is classically defined by $\chi(f) = \lim_{i \to \infty} x(i)/k$ and always exists in $F(n, n)$. It does not depend on ξ.

***Theorem** 3:* [9] Let f be a function of $F(n, n)$. The following two conditions are equivalent:

(i) There is a finite x such that $x \leq f(x)$ (respectively, $x \geq f(x)$)

(ii) $\chi(f) \geq 0$ (respectively, $\chi(f) \leq 0$) . ■

References

[1] F. Baccelli, G. Cohen, G.J. Olsder and J.P. Quadrat, *Synchronization and Linearity. An Algebra for Discrete Event Systems*, available from http://maxplus.org, New York, Wiley, 1992.

[2] P. Butkovič and G. Hegedüs, *An elimination method for finding all solutions of the system of linear equations over an extremal algebra*, Ekonomicko-Matematicky Obzor, Vol. 20, no 2, pp. 203-215, 1984.

[3] Y. Cheng and D-Z. Zheng, *Min-max inequalities and the timing verification problem with max and linear constraints*, Discrete Event Dynamic Systems : Theory and Applications, Springer Science and Business Media, Vol 15, pp. 119–143, 2005.

[4] J-P. Courtiat, M. Diaz, R.C. De Oliviera and P. Sénac, *Formal models for the description of timed behaviors of multimedia and hyper media distributed systems*, Computer Communications 19, pp. 1134-1150, 1996.

[5] B. Heidergott, G. J. Olsder and J. van der Woude, *Max Plus at Work*, Princeton University Press, 2006.

[6] P. Declerck and M.K. Didi Alaoui, *Optimal Control Synthesis in Interval Descriptor Systems, Application to Time Stream Event Graphs*, available from http://www.istia.univ-angers.fr/∼declerck, IFAC Congress Praha, 2005.

[7] P. Declerck and Didi Alaoui, M.K., *Extremal trajectories in P-time Event Graphs: application to control synthesis with specifications*, available from http://www.istia.univ-angers.fr/∼declerck, *Proc. 44th IEEE Conference on Decision and Control and European Control Conference, CDC-ECC'05*. Seville, Spain, pp.7621-7626, 2005.

[8] V. Dhingra and S. Gaubert, *How to solve large scale deterministic games with mean payoff by policy iteration*, *ACM International Conference* Proceeding Series; Vol. 180, article No.12 in the Proceedings of the 1st international conference on Performance evaluation methodolgies and tools (VALUETOOLS), Pisa, Italy, 2006.

[9] S. Gaubert and J. Gunawardena, *The duality theorem for min-max functions*, CRAS, t. 326, Série I, pp. 43-48, 1998.

[10] M. L. J. Hautus, *Controlled invariance in systems over rings*, In Feedback Control of Linear and Nonlinear Systems. Berlin: Springer-Verlag, Vol. 39, pp 107-122, 1982.

[11] R. D. Katz, *Max-Plus (A,B)-Invariant Spaces and Control of Timed Discrete-Event Systems*, IEEE Transactions on Automatic Control, Vol.52, NO. 2, pp. 229-241, February 2007.

[12] K. Mc Millan and D. Dill, *Algorithms for interface timing verification*, Proceedings of the IEEE, International Conference on Computer Design: VLSI in Computers and Processors, 1992.

[13] B. De Schutter and T. van den Boom, *Model predictive control for max-plus-linear discrete event systems*, Automatica, Vol. 37(7), pp. 1049-1056, 2001.

[14] A. Tarski, *A lattice-theoretical fixpoint theorem and its applications*, Pacific Journal of Mathematics, Vol. 5, pp. 285-309, 1955.

Ph. Declerck,

From extremal trajectories to consistency in P-time Event Graphs, IEEE Transactions on Automatic Control, Vol. 56 n°2, IETAA9, pages 463-467, février 2011.

From Extremal Trajectories to Token Deaths in P-time Event Graphs

Philippe Declerck

***Abstract*— In this paper, we consider the (max, +) model of P-time Event Graphs whose behaviors are defined by lower and upper bound constraints. The extremal trajectories of the system starting from an initial interval are characterized with a particular series of matrices for a given finite horizon. Two dual polynomial algorithms are proposed to check the existence of feasible trajectories. The series of matrices are used in the determination of the maximal horizon of consistency and the calculation of the date of the first token deaths.**

***Index Terms*— P-time Petri Nets, (max,+) algebra, token death, Kleene star, fixed point.**

I. Introduction

Petri Nets (PNs) with time can express the time behavior of Discrete Event Systems with their specifications. Two main behaviors of the transitions can be distinguished: firing as soon as possible in Timed PNs and firing in given time intervals for Time PNs. Time can be associated with places, transitions and arcs of the PNs. In Time Stream PNs, temporal intervals are associated with arcs outgoing from places and the firing interval of transitions is defined by different semantics ([5] [11]). For Timed PNs, durations can be associated with places (P-Timed PNs) or transitions (T-Timed PNs) and the relevant subclasses are equivalent. For Time PNs, temporal intervals can similarly be associated with places or transitions but the corresponding subclasses (P-Time PNs and T-Time PNs) are fundamentally different. In Time PNs, a temporal interval of firing is associated with each transition enabled by the marking while a temporal interval of availability is associated with each token which enters a place in P-Time PNs. In this paper, we focus on P-time Event Graphs [13] whose evolution can undergo token deaths which express the loss of resources or parts and failures to meet time specifications. Applications of P-time Event Graphs can be found in production systems [10], food industry [6] and transportation systems [9].

A natural aim is to characterize the trajectories followed by the system starting from an initial state. In P-time Event Graphs, it is well-known that a simple forward simulation does not guarantee the correct synchronization of the transitions and often leads to token deaths. A first objective is the determination of possible trajectories without token deaths. The concept of extremal (lowest and greatest, see [14]) trajectories is relevant for the class of P-time Event Graphs and corresponds to the earliest and latest trajectories. In this paper, the objective is to express the *relations* of these extremal trajectories from the model and the initial condition, for a given horizon.

P. Declerck is with LISA EA4014, University of Angers, 62 avenue Notre-Dame du Lac, Angers 49000, France (e-mail : philippe.declerck@univ-angers.fr).

An important notion is the consistency, which can be defined by the existence of a time trajectory following the model. A second question is to know if the different tasks can be sufficiently repeated during a period such that a given production plan can be performed. More precisely, our objective is to know if the different tasks can be repeated infinitely or during a finite period, and to determine the *maximal horizon* (maximal number of events) where the synchronizations of the transitions can be made.

A consequence is that the end of this horizon is also the limit of consistency which leads to a non-synchronization of the transitions: at least, a token death happens. The last objective is the determination of the *date of the first token deaths*.

In (max, +) algebra, other studies naturally analyse the trajectories. Using a fixed point approach, [8] considers the control of Timed Event Graphs with specifications defined by an interval model. Analysis of the consistency of interval descriptor systems as Time Stream Event Graphs is made by using the spectral vector for a given horizon while the greatest state and control trajectories are numerically calculated by an algorithm. In this paper, in-depth analysis of P-time event graphs is performed and algebraic expressions of extremal trajectories are derived. Polynomial algorithms are proposed for the determination of the maximal horizon of temporal consistency and the calculation of the first date of token deaths. This improves the pseudo-polynomial algorithm of [8] for similar problems.

Another possible approach is to rewrite the system in the form of a polyhedron in conventional algebra [6]. A priori, an application of the algorithms of linear programming can check the existence of an arbitrary trajectory. But, recall that the best algorithms of linear programming are polynomial *in the weak sense*. Contrary to these generic algorithms, we propose here algorithms specific to the considered problem whose complexity is polynomial *in the strong sense*.

If we only consider the problem of consistency, a possible technique is the model-checking which is an enumerative method based on the construction of a state class graph and its analysis. Some authors [2] apply this approach to T-Time Petri nets where each state class is defined by its marking and a set of firing times of the transitions. Starting from a given class, the firing of each enabled transition generates another class and a procedure establishes the list of the different classes and its connections. Generally speaking, model checking faces a combinatorial blow up of the state-space, commonly known as the state explosion problem, even for small systems [15]: the elementary event graph of the example of Figure 4 in [2] which is composed of two places and two transitions, illustrates this fact. As a consequence, these approaches generally consider a class of models where the graph is finite, that is, the Time Petri Nets are *bounded* and the bounds of the time intervals are defined in the *rational numbers*. In this paper, these assumptions are not taken as the considered models are non-bounded Event Graphs where the values of the temporisations are defined in $\mathbb{R}^+$. Contrary to the polyhedra of the classes which generally are approximations of the possible dates [3], the spaces considered in this paper are exact because the concept of lattice is relevant in the event graphs.

In this paper, no hypothesis is made on the structure of the Event Graphs which do not need to be strongly connected. The initial marking must only satisfy the classical condition of liveness (no circuit without token), and the usual hypothesis First In First Out (FIFO) for tokens is made.

The paper is structured as follows: We first define P-time Event Graphs and briefly introduce their algebraic model. Then, we study the time behavior of this model with the help of a special series of matrices. Notations and some previous results in the (max, +) algebra are given in Appendix I while the proof of Theorem 1 is given in Appendix II.

II. (MAX, +) MODEL OF P-TIME EVENT GRAPHS

Consider the following notations. The set of places is denoted P. The initial marking of a place $p_l \in P$ is denoted m_l. Let ${}^\bullet p_l$ denote the set of input transitions of place $p_l \in P$ and $p_l^\bullet$ the set of output transitions of p_l. Similarly, ${}^\bullet x_i$ (respectively $x_i^\bullet$) denotes the set of the input (respectively, output) places of transition x_i for $i \geq 1$. In an Event Graph, card(${}^\bullet p_l$)=card($p_l^\bullet$)=1 for each place $p_l \in P$ and we can associate only a pair (x_i, x_j) with each place $p_l \in P$, such that transition x_j is ingoing ($x_j \in^\bullet p_l$) and transition x_i is outgoing ($x_i \in p_l^\bullet$). Initial marking m_l is also associated with place p_l.

Moreover, we associate with each place $p_l \in P$ a temporal interval $[a_l, b_l]$ with $0 \leq a_l \leq b_l$ and $[a_l, b_l] \in \mathbb{R}^+ \times (\mathbb{R}^+ \cup \{+\infty\})$. Time constraints can be defined as follows. After the arrival of a token into a place p_l at time t, it is available to fire its unique downstream transition $x_j \in p_l^\bullet$ in a given interval $[t + a_l, t + b_l]$ and dies if the firing of transition x_j does not occur before $t+b_l$. In other words, the token must stay in place p_l during a duration between a_l and b_l. Before the minimal sojourn time a_l, the token is unavailable for firing transition $x_j \in p_l^\bullet$. After the maximal sojourn time b_l, the token dies.

Example.

Let us consider the P-time Event Graph of Fig. 1. The initial marking is $\begin{pmatrix} 1 & 1 & 1 & 1 \end{pmatrix}^t$ and the temporal intervals are: $[a_1, +\infty] = [3, +\infty]$, $[a_2, +\infty] = [6, +\infty]$, $[a_3, b_3] = [1, 2]$ and $[a_4, b_4] = [3, 11]$. Let us consider the following simulation for $k = 0$, 1 and 2.

k	0	1	2
x_1	4	11	11
x_2	0	7	14
x_3	0	6	13

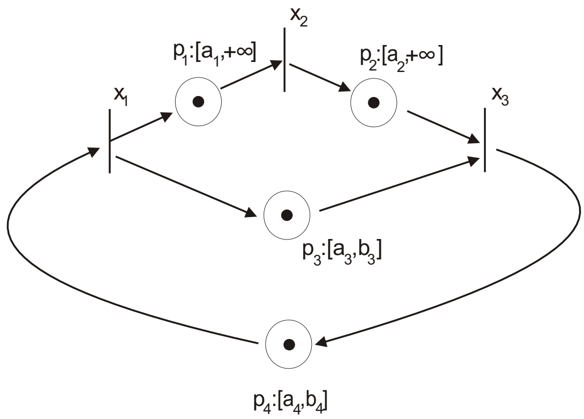

Fig. 1. P-time Event graph

k	0	1	2
p_1	$[7, +\infty]$	$[14, +\infty]$	$[14, +\infty]$
p_2	$[6, +\infty]$	$[13, +\infty]$	$[20, +\infty]$
p_3	$[5, 6]$	$[12, 13]$	$[12, 13]$
p_4	$[3, 11]$	$[9, 17]$	$[16, 24]$

The first table contains the firing dates while each column k of the second table is the bounds of the sojourn time (in absolute time) of the tokens, produced by the k^{th} firing of the transitions x_1, x_2 and x_3 in each place. We assume that the tokens of the initial marking are available immediately at $k = 0$. Let us consider the firing of transition x_3 for $k = 3$ which needs to use the tokens present in its upstream places p_2 and p_3 produced at $k = 2$. However, this synchronization does not occur because the interval $[20, +\infty] \cap [12, 13]$ is empty. A consequence is the death of the token in place p_3 at time $t = 13$.

However, transition x_1 can be fired at $t = 18$ because the interval of sojourn time of the token in place p_4 is $[16, 24]$. Therefore, a token is added in place p_3 with time interval $[19, 20]$ and the firing of transition x_3 can occur at time $t = 20$ because the interval $[20, +\infty] \cap [12, 13]$ is replaced by the interval $[20, +\infty] \cap [19, 20]$. ■

We now consider the "dater" description in the (max, +) algebra: each variable $x_i(k)$ represents the date of the k^{th} firing of transition x_i for $i \geq 1$. If we assume a FIFO functioning of the places which guarantees that the tokens do not overtake one another, a correct numbering of the events can be carried out. The evolution of the P-time Event Graph is described by the following inequalities expressing relations between the firing dates of transitions:

$\forall p_l \in P$ with $x_j \in^\bullet p_l$ and $x_i \in p_l^\bullet$, $a_l + x_j(k - m_l)$ $\leq x_i(k)$ and $x_i(k) \leq b_l + x_j(k - m_l)$

From these relations, we can derive an equivalent description of the system in (max, +) algebra [7]. By usual (max,+) algebraic notation, maximization and addition operations are denoted respectively $\oplus$ and $\otimes$. The notations and a brief review of preliminary results are presented in Appendix I. Without loss of generality, we assume that the initial marking of each place is equal to zero or one. Hence the (max, +) algebra model is as follows

$$\begin{pmatrix} x(k) \\ x(k+1) \end{pmatrix} \geq \begin{pmatrix} A^{=} & A^{+} \\ A^{-} & A^{=} \end{pmatrix} \otimes \begin{pmatrix} x(k) \\ x(k+1) \end{pmatrix} \quad (1)$$

for $k \geq 0$, where the initial condition is $x(0) = x_0$, matrix A^- (respectively, A^+) contains the temporizations a_l (respectively, b_l with minus sign) associated with each place $\forall p_l \in P$ with $m_l = 1$. We have $A^= = A_0^- \oplus A_0^+$ where matrix A_0^- (respectively, A_0^+) is defined as A^- (respectively, A^+) but with $m_l = 0$.

Example continued.

$A^= = \varepsilon$, $A^- = \begin{pmatrix} \varepsilon & \varepsilon & 3 \\ 3 & \varepsilon & \varepsilon \\ 1 & 6 & \varepsilon \end{pmatrix}$ and $A^+ = \begin{pmatrix} \varepsilon & \varepsilon & -2 \\ \varepsilon & \varepsilon & \varepsilon \\ -11 & \varepsilon & \varepsilon \end{pmatrix}$. ■

Now, we analyze the time evolution of model (1).

III. Extremal acceptable trajectories by series of matrices

Unlike a Timed Event Graph which defines a unique trajectory according to the earliest firing rule, a P-time Event Graph defines a set of trajectories which depend on matrices $A^=$, A^- and A^+. The aim of this section is to give the relations of the extremal (lowest and greatest) trajectories satisfying an initial condition given by $x(0) \in [x_0^-, x_0^+]$ (that is, box $[x_0^-, x_0^+]$ defines a set of initial conditions) and model (1) for $k = 0$, ..., $h - 1$ where $h \in \mathbb{N}$ is a finite horizon. In the sequel, we will show that these relations allow the determination of the maximal horizon of consistency (possibly infinite) and the calculation of the date of the first token deaths.

A. Expressions of the extremal state trajectories

We consider below a pair of trajectories corresponding to the earliest and greatest trajectories. The dimension of vector x is denoted n. Symbols $\wedge$ and $\backslash$ respectively correspond to the minimization operation and the left $\otimes-$ residual defined in Appendix I.

***Theorem** 1:* Given horizon $h \in \mathbb{N}$, the lowest and greatest state trajectories $(x^-(k), x^+(k)) \in ((\overline{\mathbb{R}}_{max})^n \mathrm{x} (\overline{\mathbb{R}}_{max})^n)$ for $k = 0,...,h$ respectively starting from an initial condition $x^-(0) \geq x_0^- \in (\overline{\mathbb{R}}_{max})^n$ and $x^+(0) \leq x_0^+ \in (\overline{\mathbb{R}}_{max})^n$, are given by the following equalities:

a) Coefficients of matrix w_k by forward iteration
Initialization: $w_0 = A^=$
For $k = 1$ to h, $\quad w_k = A^= \oplus A^- \otimes (w_{k-1})^* \otimes A^+$

b) First estimate (β_k^-, β_k^+) by forward iteration
Initialization: (β_0^-, β_0^+)=(x_0^-, x_0^+)
For $k = 1$ to h, (β_k^-, β_k^+)=$(A^- \otimes (w_{k-1})^* \otimes \beta_{k-1}^-$, $A^+ \backslash (w_{k-1})^* \backslash \beta_{k-1}^+)$

c) Trajectory $(x^-(k), x^+(k))$ by backward iteration
Initialization: $(x^-(h), x^+(h))$=$(\ (w_h)^* \otimes \beta_h^-, (w_h)^* \backslash \beta_h^+\ )$
For $k = h-1$ to 0, $(x^-(k), x^+(k))$=$(\ (w_k)^* \otimes [A^+ \otimes x^-(k+1) \oplus \beta_k^-], (w_k)^* \backslash [A^- \backslash x^+(k+1) \wedge \beta_k^+])$

Proof. The proof is given in Appendix II. ■

The three steps of the theorem make up two forward/backward algorithms. Identical in the calculation of the two bounds, the first step a) is the forward calculation of parameters w_k which only depends on the model. Starting from the initial condition x_0^- (respectively, x_0^+), the second step b) is also based on a forward iteration. It expresses a first estimate of the lowest (resp., greatest) trajectory denoted β_k^- (resp., β_k^+), which is finally improved by a maximisation (resp., a minimisation) in step c). The final result is the lowest (resp., greatest) trajectory denoted by $x^-(k)$ (resp., $x^+(k)$). Note that each bound can be derived from the other one by duality and each lower (resp., upper) matrix respectively corresponds to an upper (resp., lower) matrix by replacing $\oplus$ by $\wedge$, $\otimes$ by $\backslash$ and conversely.

The following property compares the intermediate trajectories (β_k^-, β_k^+) with $(x^-(k), x^+(k))$.

***Property** 1:* The pair (β_k^-, β_k^+) for $k = 0$ to $+\infty$ is a box (interval vector) containing the extremal trajectories $(x^-(k), x^+(k))$ for any finite horizon h, for a given pair of initial conditions (x_0^-, x_0^+).

Proof. Immediate: $x^-(k)$ (respectively, $x^+(k)$) is the result of a maximization (respectively, a minimization) in step c). ■

Therefore, step b) gives intermediate trajectories (β_k^-, β_k^+) for a given interval $[x_0^-, x_0^+]$, which are formally defined in the infinite horizon. As they are independant of step c), a calculation on a given finite horizon h_1 can be reused in new calculation of the extremal state trajectories for another horizon $h_2 \neq h_1$. If $h_1 < h_2$, only the calculation of (β_k^-, β_k^+) for $k = h_1 + 1, .., h_2$ is necessary.

Remark. Defined on a box $[x_0^-, x_0^+]$, the initial condition is less restrictive than the more usual $x(0) = x_0$ which is a particular case ($x_0^- = x_0^+ = x_0$). Assuming that the system is consistent, the two dual algorithms allow checking the existence of an initial condition $x(0) \in [x_0^-, x_0^+]$ in $\mathbb{R}$ which is the starting point of a finite trajectory: indeed, if A^- has no null row, trajectory x^- is finite and the check is the verification of inequality $x^-(0) \leq x_0^+$ for the lower bound $x^-(k)$. Also, the equality $x^-(0) = x_0^-$ clearly allows checking the acceptability of x_0^- or, in other words, if there is a trajectory starting from x_0^- . The same remarks hold for the dual algorithm under the condition that matrix A^+ has no null column.

B. Maximal horizon of temporal consistency

Assuming the liveness of the Event Graph, we consider the temporal consistency of P-time Event graphs. Clearly, if we can calculate an arbitrary finite trajectory (that is, in $\mathbb{R}$) starting from $x(0) \in (\mathbb{R})^n$, the system is consistent on the given horizon. Therefore, the liveness of tokens is guaranteed and it does not lead to any deadlock situation. In fact, we can prove that the existence of a finite trajectory only depends on matrices w_k and more precisely, that a finite trajectory exists if and only if matrices $(w_k)^*$ converge in $\mathbb{R}_{\max}$ [7].

Let us now consider the problem of the determination of the maximal horizon of temporal consistency. In step c) of the algorithms, the calculations of the state trajectory $x^-(k)$

start from values w_h , β_h^- and β_h^+ and consequently depend on horizon $[0, h]$ where h is a datum. Contrary to step c), the calculations of w_k, β_k^- and β_k^+ start from $A^=$, x_0^- and x_0^+ in steps a) and b): they depend on index k, but not on horizon h as the calculations can continue after h. Therefore, the problem is now to determine the maximal horizon h_{max} where the system can follow a finite trajectory. As the horizon can be finite or infinite, we consider the two following cases.

- Case 1. Matrix $(w_k)^*$ does not belong to $\mathbb{R}_{\max}$. As there is at least an infinite entry $((w_k)^*)_{i,j} = +\infty$, the system is not consistent on horizon $[0, h]$ with $h \geq k$.
- Case 2. Matrix $(w_k)^*$ belongs to $\mathbb{R}_{\max}$. If $w_k = w_{k-1}$, then the P-time Event graph is consistent on an infinite horizon and $h_{max} = +\infty$.

A practical way to determine the horizon of consistency h_{max} is as follows.

Algorithm

Initialization: $k \leftarrow 0$. Calculate and analyze $(w_k)^*$ for $k \geq 0$. Stop if case 1 ($h_{max} = k - 1$ if $k \geq 1$) or case 2 ($h_{max} = +\infty$) defined above is satisfied or repeat with $k \leftarrow k + 1$. ■

As the series w_k is non-decreasing (the proof is given in [7]), each entry converges to a stable finite value or the infinite value $+\infty$.

Example continued.

$w_0 = A^= = \varepsilon$, $w_1 = \begin{pmatrix} -8 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 1 \\ \varepsilon & \varepsilon & -1 \end{pmatrix}$, $w_2 = \begin{pmatrix} -8 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 1 \\ -4 & \varepsilon & -1 \end{pmatrix}$, $w_3 = \begin{pmatrix} -8 & \varepsilon & -3 \\ \varepsilon & \varepsilon & 1 \\ -4 & \varepsilon & 1 \end{pmatrix}$ and $(w_3)^* = \begin{pmatrix} +\infty & \varepsilon & +\infty \\ +\infty & 0 & +\infty \\ +\infty & \varepsilon & +\infty \end{pmatrix}$.

As some coefficients of w_3^* are equal to $+\infty$, $h_{max} = 2$ and a trajectory can only be defined on horizon $[0, 2]$. System (1) is only consistent for $k = 0$ and 1.

C. Date of the first token deaths

If the system is only consistent on horizon h_{max}, an admissible trajectory can be calculated but the tokens produced by the firing at date $x(h_{max})$ do not lead to a complete firing of the transitions at the following number of events $h_{max} + 1$. Below, we consider only the case of places with unitary initial marking: by reason of the lack of space, the case of places with a null initial marking is omitted but follows a similar technique. If $m_l = 1$, the time interval of token stay is $[A_{ig}^- \otimes x_g(k),\ A_{gi}^+ \backslash x_g(k]$ for a token generated par the k^{th} firing of transition g in a place $p_l \in P$ such that $x_g \in^\bullet p_l$ and $x_i \in p_l^\bullet$. As there is at least one transition i such that relation $\bigoplus_{j\in^\bullet(^\bullet x_i)} A_{ij}^- \otimes x_j(h_{max}) \leq x_i(h_{max}+1) \leq \bigwedge_{j\in^\bullet(^\bullet x_i)} A_{ji}^+ \backslash x_j(h_{max})$ is not satisfied, the non-synchronization of transition i leads to some token deaths. Let G be the set of transitions $g \in^\bullet (^\bullet x_i)$ such that $A_{gi}^+ \backslash x_j(h_{max}) = \bigwedge_{j\in^\bullet(^\bullet x_i)} A_{ji}^+ \backslash x_j(h_{max})$. Each input transition $g \in G$ generates a token which dies in the place $p_l \in (x_g)^\bullet \bigcap {}^\bullet(x_i)$ at the date $A_{gi}^+ \backslash x_j(h_{max})$.

However, the firing of transition i is still possible if a new firing of each transition $g \in G$ produces another token. This can be expressed by a shift in the numbering of the events. Therefore, relation $A_{ig}^- \otimes x_g(k) \leq x_i(k+1) \leq A_{gi}^+ \backslash x_g(k)$ for $k < h_{max}$, becomes relation $A_{ig}^- \otimes x_g(k+1) \leq x_i(k+1) \leq A_{gi}^+ \backslash x_g(k+1)$ for $k \geq h_{max}$, in the new algebraic model.

Example continued.

For $x_0^- = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}^t$, Theorem 1 provides the lowest trajectory x^- which is also trajectory x given in the first table of the example in part II. Using these dates, we can deduce that the date of the first token death is 13. The new model is as follows: for $k \geq 2$, matrices $B^=$, B^- and B^+, replace the previous one in system (1).

$B^= = \begin{pmatrix} \varepsilon & \varepsilon & -2 \\ \varepsilon & \varepsilon & \varepsilon \\ 1 & \varepsilon & \varepsilon \end{pmatrix}$, $B^- = \begin{pmatrix} \varepsilon & \varepsilon & 3 \\ 3 & \varepsilon & \varepsilon \\ \varepsilon & 6 & \varepsilon \end{pmatrix}$ and $B^+ = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ -11 & \varepsilon & \varepsilon \end{pmatrix}$. ■

IV. CONCLUSION

Considering the (max, +) model of P-time Event Graphs, our first objective is the determination of the extremal state trajectories satisfying an initial condition defined on an interval. Based on a specific series of matrices, the proposed resolution is composed of three steps: using the Kleene star, the iterative calculation determines the values of the greatest paths for different horizons; a forward iteration generates a box containing the extremal trajectories; a backward iteration gives the extremal trajectories. The introduction of a nondecreasing series of matrices alleviates the storage as the dimension is the size of the model, which depends on the number of the transitions and the initial marking. Therefore, each calculation processes reduced matrices of dimension $(n \times n)$. The approach can be applied to important processes for large horizons because the algorithms are strongly polynomial: the complexity is $O(h.n^3)$ for a given horizon h if the complexity of the used algorithm of Kleene star is $O(n^3)$ ([7] gives the CPU time for different dimensions and horizons).

The determination of the maximal horizon of temporal consistency is the second objective. The technique is based on the analysis of convergence of matrices w_k^*: each entry can converge to a stable finite value or the infinite value $+\infty$. For a given P-time Event Graph, the case of a convergence to a constant matrix after a transitory period h_{max}, facilitates the storage and the reuse in the calculation of a new trajectory for any horizon. If the system is only consistent on horizon h_{max}, a non-synchronization cannot be avoided at $h_{max} + 1$ and we calculate the date of the first token deaths.

APPENDIX I

In this section, we shall review a few basic theoretical notions about dioids. For more extensive presentations, the reader is invited to consult the following reference [1].

A monoid is a couple $(S, \oplus)$ where operation $\oplus$ is associative and presents a neutral element. A semi-ring S is a triplet $(S, \oplus, \otimes)$ where $(S, \oplus)$ and $(S, \otimes)$ are monoids, $\oplus$

is commutative, $\otimes$ is distributive in relation to $\oplus$ and the zero element ε of $\oplus$ is the absorbing element of $\otimes$ ($\varepsilon \otimes a = a \otimes \varepsilon = \varepsilon$). A dioid D is an idempotent semi-ring (operation $\oplus$ is idempotent, that is $a \oplus a = a$). Let us note that unlike the structures of group and ring, monoids and semi-rings do not have a property of symmetry on S. The set $\mathbb{R} \cup \{-\infty\}$ provided with the maximum operation denoted $\oplus$ and the addition denoted $\otimes$ is an example of dioid which is usually denoted $\mathbb{R}_{max}$ = $(\mathbb{R} \cup \{-\infty\}, \oplus, \otimes)$. The neutral elements of $\oplus$ and $\otimes$ are represented by $\varepsilon = -\infty$ and $e = 0$, respectively.

A dioid D is complete if it is closed for infinite sums and the distributivity of the multiplication with respect to addition extends to infinite sums: $(\forall\ c \in D\ )\ (\forall\ A \subseteq D)\ c \otimes (\bigoplus_{x \in A} x) = \bigoplus_{x \in A} c \otimes x$. For example, $(\mathbb{R} \cup \{-\infty\} \cup \{+\infty\}, \oplus, \otimes)$ usually denoted $\overline{\mathbb{R}}_{max}$, is complete. The set of nxn matrices with entries in a complete dioid D included with the two operations $\oplus$ and $\otimes$ is also a complete dioid, which is denoted D^{nxn}. Nonsquare matrices can be considered if they are completed with rows or columns with entries equal to ε. The sum and product of matrices are defined conventionally from the sum and product in D.

Let Γ be a subset of vectors over $\overline{\mathbb{R}}_{max}$. The partial natural order denoted $\leq$ is defined as follows: $v \leq w \Longleftrightarrow v \oplus w = w$. It is also a componentwise order which allows the comparison of any pair of vectors (v, w) i.e. $v \leq w \Longleftrightarrow v_i \leq w_i$, for each component i. In the paper, this concept is applied to control and state trajectories. The element v of subset Γ is called greatest element or maximum element if and only if $w \leq v$ for all $w \in \Gamma$. In other words, it is greater than any other element of the subset: (see part 4.3.1 of [1] for more details). If this greatest element exists, it is unique as the existence of two different maximum elements v and w implies $w \leq v$ and $v \leq w$. Let $v \wedge w$ denote the lower bound of v and w.

A mapping f is monotone or isotone if $x \leq y$ implies $f(x) \leq f(y)$. Let f: $E \rightarrow F$ be an isotone mapping, where $(E, \leq)$ and $(F, \leq)$ are ordered sets. Mapping f is said to be residuated if for all $y \in F$, the least upper bound of subset $\{x \in E \mid f(x) \leq y\}$ exists and belongs to this subset. The corresponding mapping, denoted $f^d(y)$ is called the residual of f. When f is residuated, f^d is the only isotone mapping, such that $f \circ f^d \leq Id_F$ and $f^d \circ f \geq Id_E$ where Id_F and Id_E are identity mappings. Mapping $x \in (\overline{\mathbb{R}}_{max})^n \mapsto A \otimes x$, defined over $\overline{\mathbb{R}}_{max}$ is residuated [1] and the left $\otimes-$ residual of b by A is denoted by: $A \backslash b = \max\{x \in (\overline{\mathbb{R}}_{max})^n$ such that $A \otimes x \leq b\}$. Moreover, $(A \backslash b)_i = \bigwedge_{j=1}^{m} A_{ji} \backslash b_j$ where A is an $m \times n$ matrix. Using the Kleene star defined by: $A^* = \bigoplus_{i=0}^{+\infty} A^i$, the following theorem will be considered in the dioid of matrices.

***Theorem** 2:* (Theorem 4.75 part 1 in [1]) Consider equation $x = A \otimes x \oplus B$ and inequality $x \geq A \otimes x \oplus B$ with A and B in complete dioid D. Then, $A^* \otimes B$ is the least solution of these two relations. ■

APPENDIX II

Proof. System (1) for $k = 0, \ldots, h-1$ with $x(0) \geq x_0^-$ can be rewritten as follows.

$$\begin{cases} x(0) \geq A^= \otimes x(0)\ \oplus A^+ \otimes x(1) \oplus x_0^- \\ x(k) \geq A^- \otimes x(k-1) \oplus A^= \otimes x(k)\ \oplus A^+ \otimes x(k+1) \\ \text{ for } k = 1 \text{ to } h-1 \\ x(h) \geq A^- \otimes x(h-1) \oplus A^= \otimes x(h) \end{cases} \tag{2}$$

Theorem 2 shows that the smallest solution to this system also satisfies the corresponding equality and we can now consider the above system with equalities. The following proposition $\mathcal{P}(k)$ is now proved by recursion.

$\mathcal{P}(k)$: $x^-(k) = (w_k)^* \otimes [A^+ \otimes x^-(k+1) \oplus \beta_k^-]$

Base case: $\mathcal{P}(0)$

From the first equality of (2), we can write $x(0) = w_0 \otimes x(0) \oplus A^+ \otimes x(1) \oplus \beta_0^-$ where $w_0 = A^=$ and $\beta_0^- = x_0^-$. Therefore, $x(0) = (w_0)^*[A^+ \otimes x(1) \oplus \beta_0^-]$, which proves $\mathcal{P}(0)$.

Case: $\mathcal{P}(1)$

From the second equality of (2), we can write for $k = 1$, $x(1) = A^= \otimes x(1) \oplus A^- \otimes x(0) \oplus A^+ \otimes x(2)$. If $\mathcal{P}(0)$ is used, $x(1)$=$A^= \otimes x(1) \oplus A^- \otimes [(w_0)^*[A^+ \otimes x(1) \oplus \beta_0^-]] \oplus A^+ \otimes x(2)$. The distributivity of $\otimes$ with respect to $\oplus$ leads to $x(1) = [A^= \oplus A^- \otimes (w_0)^* \otimes A^+] \otimes x(1) \oplus A^- \otimes (w_0)^* \otimes \beta_0^- \oplus A^+ \otimes x(2) = w_1 \otimes x(1) \oplus \beta_1^- \oplus A^+ \otimes x(2)$ where $w_1 = A^= \oplus A^-(w_0)^* A^+$ and $\beta_1^- = A^-(w_0)^* \otimes \beta_0^-$. Therefore, $x(1) = (w_1)^* \otimes [A^+ \otimes x(2) \oplus \beta_1^-]$ and $\mathcal{P}(1)$ is proved. Now, this approach is generalized for $k = 1$ to $h-1$.

Case: $\mathcal{P}(k)$ for k from 1 to $h-1$.

Let us assume $\mathcal{P}(k-1)$: $x(k-1) = (w_{k-1})^* \otimes [A^+ \otimes x(k) \oplus \beta_{k-1}^-]$. We will prove that $\mathcal{P}(k-1)$ entails $\mathcal{P}(k)$. From the second equality of (2), we can write $x(k) = A^= \otimes x(k) \oplus A^- \otimes x(k-1) \oplus A^+ \otimes x(k+1)$. As $x(k-1) = (w_{k-1})^* \otimes [A^+ \otimes x(k) \oplus \beta_{k-1}^-]$, the following expression is deduced: $x(k) = A^= \otimes x(k) \oplus A^- \otimes (w_{k-1})^* \otimes [A^+ \otimes x(k) \oplus \beta_{k-1}^-] \oplus A^+ \otimes x(k+1)$. The distributivity of $\otimes$ with respect to $\oplus$ yields $x(k) = [A^= \oplus A^- \otimes (w_{k-1})^* \otimes A^+] \otimes x(k) \oplus A^- \otimes (w_{k-1})^* \otimes \beta_{k-1}^- \oplus A^+ \otimes x(k+1) = w_k \otimes x(k) \oplus \beta_k^- \oplus A^+ \otimes x(k+1)$, where $w_k = A^= \oplus A^- \otimes (w_{k-1})^* \otimes A^+$and $\beta_k^- = A^- \otimes (w_{k-1})^* \otimes \beta_{k-1}^-$. Therefore, $x(k) = (w_k)^*[A^+ \otimes x(k+1) \oplus \beta_k^-]$ and the desired expression is obtained: $\mathcal{P}(k)$ has been deduced from $\mathcal{P}(k-1)$. Moreover, as $\mathcal{P}(0)$ is true, $\mathcal{P}(k)$ has been proved for k from 1 to $h-1$: the recursion is finished. Knowing β_k^-, the calculation of $x(k)$ uses a backward iteration, while the calculation of β_k^- is relevant to a forward iteration. Now, the final case will be proved.

Case: $\mathcal{P}(h)$

The last equality of (2) can be considered like the second equality but without $A^+ \otimes x(k+1)$: the argument of case $\mathcal{P}(k)$ can be taken and we can write $x(h) = (w_h)^* \otimes \beta_h^-$ with $w_h = A^= \oplus A^- \otimes (w_{h-1})^* \otimes A^+$and $\beta_h^- = A^- \otimes (w_{h-1})^* \otimes \beta_{h-1}^-$

The proof of the greatest trajectory is omitted as it can be deduced by duality from the previous proof. Indeed, as mapping $A^= \otimes x(k)$, $A^- \otimes x(k-1)$ and $A^+ \otimes x(k+1)$ are residuated, the application of property f3 in [1] part 4.4.4) gives the following form: it expresses every "upper" constraint on $x(k)$ which can minimize it.

$$\begin{cases} x(0) \leq A^= \backslash x(0) \wedge A^- \backslash x(1) \wedge x_0^+ \\ x(k) \leq A^+ \backslash x(k-1) \wedge A^= \backslash x(k) \wedge A^- \backslash x(k+1) \\ \text{ for } k = 1 \text{ to } h-1 \\ x(h) \leq A^+ \backslash x(h-1) \wedge A^= \backslash x(h) \quad \blacksquare \end{cases}$$

References

[1] F. Baccelli, G. Cohen, G.J. Olsder and J.P Quadrat, *Synchronization and linearity. An Algebra for Discrete Event Systems*, available from http://maxplus.org, New York, Wiley, 1992.

[2] B. Berthomieu and M. Diaz, Modeling and verification of time dependent systems using time Petri nets, IEEE Transactions on Software Engineering, 17(3), 1991.

[3] B. Berthomieu, D. Lime, O.H. Roux and F. Vernadat, Reachability Problems and Abstract State Spaces for Time Petri Nets with Stopwatches, Journal of Discrete Event Dynamic Systems, No. 17, pp. 133-158, 2007

[4] M. Boyer and O.(H.) Roux, On the compared expressiveness of arc, place and transition time Petri nets, Fundamenta Informaticae, No. 88(3), pp 225-249, 2008.

[5] J.P. Courtiat, M. Diaz, R.C. De Oliviera and P. Sénac, Formal models for the description of timed behaviors of multimedia and hyper media distributed systems, *Computer Communications,* No. 19, pp.1134-1150, 1996.

[6] P. Declerck, A. Guezzi and C. Gros. *Temps de cycle des Graphes d'Evénements Temporisés et P-temporels*. Conférence Internationale Francophone d'Automatique, CIFA'2008, Bucharest, Romania, 2008.

[7] P. Declerck, From extremal trajectories to consistency in P-time Event Graphs, available from http://www.istia.univ-angers.fr/∼declerck/, technical report, Lisa, University of Angers, September 26, 2007.

[8] P. Declerck and M.K. Didi Alaoui, Optimal control synthesis of Timed Event Graphs with specifications defined by an interval model, *IEEE Transactions on Automatic Control*, February 2010.

[9] F. Defossez, S. Collart-Dutilleul and P. Bon, *Temporal requirements checking in a safety analysis of railway systems*, FORMS/FORMAT 2008, Symposium on Formal Methods for Automation and Safety in Railway and Automotive Systems, TU Braunschweig and Budapest University of Technology and Economics, October, 2008.

[10] F. Chetouane, J.P. Denat and S. Collart-Dutilleul, *A flexible control for manufacturing automated electroplating lines*, International Journal for Manufacturing Science & Production, Vol. 8, No. 1, pp. 33-48, 2007.

[11] M. Diaz and P. Owezarski, From multimedia models to multimedia transport protocols, *Computer Networks and ISDN Systems*, No. 29, pp. 745-758, 1997.

[12] M.K. Didi Alaoui, Etude et supervision des graphes d'événements temporisés et temporels : vivacité, estimation et commande, available from http://www.istia.univ-angers.fr/∼declerck/, *PhD Thesis*, University of Angers, France, 2005.

[13] W. Khansa, J-P. Denat and S. Collart Dutilleul, P-time Petri nets for manufacturing systems, *WODES'96*, Edinburgh, UK, 1996.

[14] R. Kumar, and V.K. Garg, Extremal solutions of inequations over lattices with applications to supervisory control, *Theoret. Comp. Sci.*, No. 148, pp. 67-92, 1995.

[15] M. Magnin. Réseaux de Petri à chronomètres : temps dense et temps discret, *PhD Thesis*, Ecole Centrale de Nantes, France, December 2007.