Projet LAAS

Ordonnancement dans les Systèmes Embarqués Critiques (OSEC)
-
Incertitude sur la Demande et Robustesse

29 janvier 2010

Coordonnateur du projet : Olivier Brun (MRS)

Responsables scientifiques :
- Christian Artigues (MOGISA)
- Olivier Brun (MRS)
- Pierre-Emmanuel Hladik (OLC)

Groupes de recherches impliqués :
- « Modélisation, Optimisation et Gestion Intégrée de Systèmes d'Activités » (MOGISA)
- « Modélisation et controle des Réseaux et Signaux » (MRS)
- « Outils Logiciels pour la Communication » (OLC)
- « Stochastic Operations Research », TU Eindhoven, Pays-Bas.

Résumé : Le bon fonctionnement d'un système temps réel dépend non seulement de l'exécution correcte des tâches, mais aussi du respect de leurs contraintes temporelles. La conception de ces systèmes s'appuie souvent sur une analyse pessimiste basée sur les temps d'exécution au pire pour construire un ordonnancement satisfaisant strictement les exigences temps réel des tâches. Cette analyse conduit généralement à un surdimensionnement excessif des ressources, et impose des contraintes très fortes sur les développements. Si elle est appropriée dans certains domaines comme l'aéronautique, cette approche n'est pas du tout adaptée dans d'autres contextes applicatifs. Avec une modélisation plus fine, prenant en compte l'incertitude sur les temps d'exécution, et en se contentant de garanties probabilistes sur le respect de l'échéance pour les tâches ayant des exigences temp réel souples, il serait possible de réduire drastiquement le dimensionnement de l'architecture et de simplifier les développements. De même, l'intégration des besoins de reconfiguration en cas de panne dès la conception de l'ordonnancement doit permettre des gains très importants.

Durée du projet : 2 ans.

Budget : 94328 euros.

Évaluateurs externes potentiels du projet :
- Frédéric Boniol, ONERA (frederic.boniol@onera.fr).
- Claire Hanen, Professeur Université de Nanterre / LIP6 (claire.hanen@lip6.fr)
- Eric Sanlaville, Professeur à l'Université du Havre / LITIS (Eric. Sanlaville@univ-lehavre.fr)

1 Objectifs scientifiques

Mots clés : Systèmes embarqués critiques, temps réel, ordonnancement, vérification, incertitudes, reconfiguration

Ces dernières années, les systèmes embarqués critiques ont connu des évolutions importantes. Face à l'explosion du nombre des fonctions qu'ils réalisent, et à une augmentation forte de la complexité globale (grand nombre de calculateurs et de bus imbriqués), de nouvelles architectures ont vu le jour. Celles-ci ont permis une homogénéisation des plate-formes embarquées (IMA pour l'avionique et ECU pour l'automobile) et une standardisation des supports d'exécution (ARINC en avionique et AUTOSAR en automobile). Cependant ces évolutions ont reporté la complexité vers la phase d'intégration des fonctions et de leurs communications. Pour pallier cela, il devient nécessaire de fournir des outils performants pour guider les différents acteurs au cours du cycle de développement de ces systèmes.

Parmi les gains apportés par les nouvelles architectures, la diminution du nombre de ressources embarquées (par la possibilité offerte de multiplexer ces ressources) et donc la diminution du poids, se traduit par la nécessité de partager temporellement et spatialement une ressource par un plus grand nombre de fonctions. Les choix qui pouvaient être effectués « manuellement » par un expert pour dimensionner un système, sont devenus irréalisables de par la taille des problèmes considérés et la complexité des interactions entre les fonctions. La problématique de l'allocation et de l'ordonnancement pour les systèmes embarqués devient une réalité à laquelle les industriels du domaine des systèmes embarqués sont confrontés et pour laquelle il n'existe pas de réponse simple.

Ce qui distingue l'ordonnancement (affectation temporelle et spatiale) des systèmes embarqués critiques par rapport aux problèmes, par exemple, de production ou de tournée de véhicule, réside dans l'aspect récurrent des fonctions à réaliser, dans la garantie forte des contraintes et dans les mécanismes d'ordonnancement dynamiques sur lesquels ils s'appuient. De nombreux travaux ont déjà abordé ce thème, mais les nouveaux standards ont aussi amené de nouvelles contraintes et critères à prendre en considération. De même, l'évolution des supports d'exécution, comme la généralisation des supports multicoeurs, introduit de nouvelles problématiques auxquels les techniques existantes ne sont pas adaptées.

Dans ce contexte, tenir compte des incertitudes au cours de la conception et de la vérification des ordonnancements pour les systèmes embarqués critiques est à notre connaissance peu considéré. Ces incertitudes peuvent être dues aussi bien à la connaissance que l'on a des caractéristiques du système, qu'aux aléas survenant pendant son fonctionnement. Par exemple, ne considérer que le pire temps d'exécution pour estimer les durées des fonctions conduit à des surestimations de l'occupation des ressources. Une meilleure prise en compte des incertitudes sur les temps d'exécution et l'intégration de garanties probabilistes sur le respect des échéances de certaines tâches, doit permettre un dimensionnement beaucoup plus fin qu'une analyse dans le pire des cas, et simplifier grandement le développement des systèmes temps réel. De même, l'intégration des besoins de reconfiguration en cas de panne dès la conception de l'ordonnancement doit permettre des gains très importants. Les enjeux applicatifs de ces recherches sont considérables.

En réunissant des compétences en ordonnancement temps réel, modélisation stochastique et recherche opérationnelle, le projet a pour ambition de développer de nouvelles approches métho-

dologiques pour la conception des systèmes embarqués temps réel. Il s'agit d'établir une synergie entre des chercheurs ayant tous une expertise reconnue sur la problématique de l'ordonnancement, mais considérée dans des contextes différents (ordonnancement temps réel, ordonnancement stochastique et ordonnancement des systèmes de production), pour dépasser les limites des approches actuellement utilisées dans la conception des systèmes embarqués.

1.1 État de l'art

1.1.1 Ordonnancement temps réel

En raison de la nature critique de la majorité des applications embarquées temps réel, la garantie a priori des diverses exigences de ces systèmes est obligatoire. Cela nécessite le développement de techniques et d'outils permettant d'assurer la conformité du système vis-à-vis de sa spécification. Ceux-ci doivent s'inscrire dans un processus de conception et doivent permettre de valider aussi bien les aspects fonctionnels que extra-fonctionnels. Parmi ces techniques, la production et la validation des ordonnancements au niveau spatial (allocation des fonctions sur les ressources) et temporel (allocation des fonctions dans le temps) est une nécessité.

En général, deux approches sont distinguées dans le domaine de l'ordonnancement temps réel. La première, qualifiée d'ordonnancement dirigé par le temps, consiste à construire un séquencement des fonctions à exécuter, c'est-à-dire que des séquences infinies d'éléments du type (date, identifiant fonction) sont produites pour chaque ressource d'exécution et seront déroulées pendant la vie du système. Les séquences jouées doivent tenir compte des caractéristiques des fonctions (durée d'exécution, période, etc.) et garantir le respects des contraintes temporelles (échéances, gigue, etc.) ou fonctionnelle (précédence, synchronisation, etc.). Les techniques utilisées pour construire ces séquences sont variées [9] (méthodes ad-hoc, programmation linéaire, branch-and-bound, etc.). La complexité de ces techniques réside dans la capacité à prendre en considération un grand nombre de contraintes et à les intégrer dans une méthode de recherche.

La seconde approche, qualifiée d'ordonnancement dirigé par les événements, consiste à définir des règles d'ordonnancement qui réagissent dynamiquement à la demande d'exécution des fonctions. Ces règles permettent d'en élire une fonction parmi celles à exécuter et de lui attribuer les ressources nécessaires. En général, les mécanismes de choix sont assez simples (priorité, urgence, etc.). Le problème de l'ordonnancement consiste alors à vérifier a priori que toutes les exécutions possibles respectent les contraintes. Cette vérification peut s'effectuer soit par simulation [18], soit à l'aide de méthodes analytiques [9] ou par des méthodes formelles [36]. La difficulté de telles techniques réside dans la prise en considération de la complexité induite par les mécanismes dynamiques et par la granularité à laquelle les fonctions sont modélisées.

Pour les deux types d'approche, la prise en considération d'incertitudes, aussi bien sur les caractéristiques des fonctions que sur les aléas de fonctionnement, est assez peu étudiée dans la communauté temps réel et utilise en général des méthodes ad-hoc.

1.1.2 Ordonnancement stochastique

Dans de nombreuses applications, les activités à ordonnancer sont soumises de manière aléatoire et ont des temps d'exécution non prévisibles. Cela suggère l'étude de ces problèmes dans le cadre de la théorie des files d'attente. Une abondante littérature a été consacrée à l'ordonnancement optimal dans les systèmes à un seul serveur (modélisant la ressource). Un résultat d'optimalité bien connu concerne la politique SRPT (Shortest Remaining Processing Time), qui

à tout moment sert le client avec le temps de service restant le plus petit [38, 39, 41].

SRPT repose sur la connaissance des temps d'exécution. Dans la mesure où cette information n'est parfois pas disponible, d'autres travaux de recherche se sont concentrés sur les politiques non-anticipatives, les plus connues étant FCFS (First Come First Served), LAS (Least Attained Service) [34], PS et DPS. Ces politiques n'utilisent aucune information sur les temps d'exécution, mais gardent trace du temps alloué à chaque activité présente dans le système. Parmi les politiques non anticipatives, celle minimisant le nombre moyen d'activités est obtenu par la régle de Gittins [3, 22]. Pour certaines distributions des temps d'exécution, cette politique correspond simplement à LAS [3, 37] ou FCFS [37]. Signalons enfin la régle $c\mu$ qui permet de minimiser l'espérance d'une somme pondéree du nombre d'activités pour de nombreux systèmes [10, 21, 33].

Si de nombreux travaux ont été consacrés à l'ordonnancement stochastique, très peu prennent en compte les caractéristiques propres des systèmes temps réel : ordonnancement cyclique des activités et contraintes sur les délais de traitement. Des extensions à la théorie des files d'attente ont été proposées dans [28, 30, 29, 15, 27, 6], tandis que d'autres auteurs ont proposé l'utilisation de techniques de comparaison stochastique [32, 13, 31]. Ces travaux sont relativement récents dans le domaine, émanent souvent de chercheurs venant de la communauté temps réel, et non pas encore, à notre connaissance, été appliqués pour concevoir des systèmes réels. La théorie est encore embryonnaire. Un des objectifs du projet est de participer à son développement en s'appuyant sur nos compétences en modélisation stochastique.

1.1.3 Ordonnancement cyclique et systèmes à événements discrets

L'évolution d'un ordonnancement cyclique se traduit par l'apparition de certains événements tels que le début ou la fin d'une activité. En considérant ce caractère événementiel, les problèmes d'ordonnancement cyclique peuvent être approchés à travers la théorie des Systèmes Dynamiques à Événements Discrets (SDED). Cette théorie pouvant traduire des phénomènes comme le parallélisme, le décalage événementiel, le décalage temporelle ou encore le partage de ressource trouve naturellement sa place dans l'étude des problèmes d'ordonnancement et *a fortiori* d'ordonnancement cyclique. Ces apports vont de la commande, qui dans le cadre de l'ordonnancement peut se traduire par la date de début des tâches ou encore par la mise à disposition de certaines ressources, à l'évaluation de performances.

On peut citer deux approches "système" courantes pour caractériser cette classe de problème. Tout d'abord, l'approche par Réseaux de Petri (RdP). Les RdP sont un langage graphique de description des phénomènes de concurrence et de synchronisation. La version temporisée des réseaux de Pétri est particulièrement bien adaptée pour transcrire le fonctionnement des ordonnancements cycliques [26],[11].

Une autre approche découle d'une sous classe de RdP temporisés ne pouvant traduire que les phénomènes de synchronisation (et donc de retard) : les graphes d'événements temporisés. Le fonctionnement dynamique de ces derniers peut s'exprimer sous la forme d'une représentation élegante dans l'algèbre des dioïdes ($(min, +)$ ou $(max, +)$). Cette description des SDED peut s'appliquer à la modélisation des ordonnancements cycliques tant dans le domaine deterministe [23] que dans le domaine stochastique [35].

1.1.4 Ordonnancement cyclique en recherche opérationnelle

Dans la théorie de l'ordonnancement déterministe, l'ordonnancement cyclique est un problème d'optimisation combinatoire consistant à déterminer une exécution périodique réalisable ou optimale d'un ensemble de tâches soumises à des contraintes de succession et de limitation

de ressources. Les caractéristiques des tâches (durées, demandes en ressources) sont supposées connues a priori. Dans les systèmes informatiques, les tâches correspondent à des instructions de programmes ou des à processus, les successions entre tâches sont liées à des dépendances (correspondant par exemple à des échanges de données) et les ressources, disponibles en quantités limités, correspondent à de la mémoire, des processeurs, des registres, etc.

Formellement, un ensemble fini de tâches génériques $\{o_1, \ldots, o_n\}$ doit être exécuté de manière périodique. Chaque tâche générique o_i a une infinité de réalisations (ou instances) $o_{i,k}$, $k \geq 1$. Un ordonnancement t donne une date de début $t_{i,k}$ à chaque instance $o_{i,k}$. Un ordonnancement est K-périodique s'il existe un entier K (la périodicité) et une valeur P (la période) tels que $t_{i,k+K} = t_{i,k} + P$. La figure 1 présente un ordonnancement de 6 instances de $n = 4$ tâches génériques, de périodicité $K = 2$ et de période $P = 5$. Le *gabarit* d'ordonnancement est le motif qui se répète du fait du caractère cyclique de l'ordonnancement.

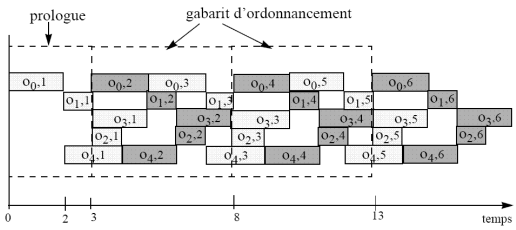

FIG. 1 – Ordonnancement 2-périodique de période 5.

Le débit d'un ordonnancement K-périodique est donné par K/P. Compte tenu de contraintes de dépendances (successions imposées entre les tâches) et de ressources (qui limitent le nombre de tâches exécutables en parallèle), on peut chercher, pour une périodicité K donnée, à maximiser le débit (donc à minimiser la période P) ou "simplement" à vérifier qu'un ordonnancement d'une certaine période P existe. Le problème est polynomial en l'absence de contraintes de ressources (il se ramène à la recherche d'un circuit critique dans le graphe des contraintes de précédences) mais NP-difficile en présence de contraintes de ressources. De nombreux travaux ont été dédiés à la résolution exacte et approchée de ce problème [8, 12, 24].

Toutefois, la mise en œuvre pratique de cette approche a des limites connues en ce qui concerne la prise en compte des incertitudes. Lorsque des événements imprévus surviennent, l'ordonnancement prévu ne peut parfois plus être exécuté et il est nécessaire de recalculer un ordonnancement. Dans les systêmes embarqués critiques, le temps nécessaire à ce calcul peut ne pas être disponible. L'objectif de notre projet est de traiter ce problême sans nécessairement surdimentionner le systême pour pallier cette difficulté.

2 Approches et méthodes envisagées

Le projet aborde plusieurs problématiques couvrant l'ordonnancement des systèmes embarqués critiques avec incertitudes :

- Conception des systèmes embarqués temps réel avec incertitude sur les temps d'exécution et garanties probabilistes sur le respect des échéances,

– Reconfiguration de l'ordonnancement des architectures distribuées temps réel en cas de défaillance matérielle.

2.1 Tâche A : Étude de cas

Deux études de cas sont envisagées pour le projet. La première a pour cadre les systèmes embarqués dans l'avionique et se base sur le standard ARINC [2]. Le but est d'étudier l'ordonnancement des partitions (structure permettant d'assurer la ségrégation spatiale et temporelle des applications avioniques) et l'impact que produit la reconfiguration [40, 7] sur celui-ci.

La seconde étude sera une application de contrôle moteur construite sur le standard AUTOSAR [1]. Dans ce cas, ne considérer que le pire cas peut conduire à une charge globale du système largement supérieure à 100%. L'hypothèse du pire cas est alors impossible à exploiter et il devient nécessaire d'avoir une représentation stochastique de l'application pour en faire l'analyse.

Ces deux études de cas permettront de formaliser les modèles utilisés dans les différentes tâches et d'identifier clairement les objectifs. De plus, elles serviront à démontrer la faisabilité et la pertinence des méthodes proposées.

2.2 Incertitude sur la demande

Le bon fonctionnement d'un système temps réel dépend non seulement de l'exécution correcte des tâches, mais aussi du respect de leurs contraintes temporelles. La conception de ces systèmes s'appuie souvent sur une analyse dans le pire des cas pour construire un ordonnancement satisfaisant strictement les exigences temps réel des tâches[1]. Cette analyse est pessimiste car elle repose sur les temps d'exécution dans le pire des cas des tâches (wcet : worst-case execution time). Elle conduit donc généralement à un surdimensionnement excessif des ressources, et impose des contraintes très fortes sur les développements. Par exemple, elle peut conduire à supposer qu'un traitement qui ne se produit que de manière exceptionnelle, est en fait effectué régulièrement. Elle est néanmoins appropriée dans certains domaines applicatifs, comme par exemple l'aéronautique, où les contraintes de sécurité sont primordiales.

Dans d'autres domaines, comme par exemple celui de l'automobile, cette approche n'est pas adaptée. Une modélisation plus fine, prenant en compte l'incertitude sur les temps d'exécution, peut permettre de réduire le surdimensionnemnt et simplifier les développements. D'autre part, alors que certaines tâches peuvent avoir des exigences temps réel strictes, d'autres peuvent avoir des contraintes plus souples : une garantie probabiliste sur le respect de l'échéance peut être suffisante. Un des objectifs du projet est de développer des outils théoriques pour la conception des systèmes temps réel, prenant en compte ces deux aspects. Les travaux qui seront conduits sont décrits ci-dessous.

2.2.1 Tâche B1 : Évaluation de l'ordonnançabilité par file d'attente

Considérons l'ordonnancement périodique d'un ensemble $\mathcal{T} = \{1, \ldots, N\}$ de N tâches temps réel indépendantes sur un processeur. Ce système peut-être modélisé par une file d'attente temps réel comme décrit sur la Figure 2.

[1] O. Brun et P.-E. Hladik collabore déja dans le cadre du projet SATRIMMAP sur l'ordonnancement de fonctions avioniques sur une architecture IMA. Les travaux développés sont basés sur une analyse dans le pire des cas. Le projet OSEC s'appuie sur cette expérience, mais adresse des problématiques différentes.

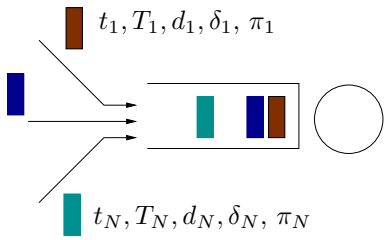

FIG. 2 – File d'attente à plusieurs classes de trafic.

Les tâches de classe-i arrivent selon un processus déterministe de période T_i, la première arrivant à la date t_i. Pour prendre en compte les aléas sur les durées d_i d'exécution des tâches de classe i, nous supposons que celles-ci sont i.i.d. selon la loi de probabilités $\mathcal{D}_i$. Une tâche de classe-i a une échéance relative δ_i, c'est à dire que son exécution doit être terminée au plus tard δ_i unités de temps après son arrivée. Enfin, on suppose que le concepteur du système souhaite qu'une proportion des tâches d'au plus π_i soient hors délai, ce qui donne une garantie probabiliste sur les performances du système (une garantie stricte est obtenue avec $\pi_i = 0$). On notera S_i la durée aléatoire de l'exécution d'une tâche de classe i à l'équilibre.

Nous supposerons que la distribution $\mathcal{D}_i$ est à support borné pour toute tâche i, c'est à dire qu'il existe un temps d'exécution au pire D_i tel que $Pr\left[d_i > D_i\right] = 0$. Pour certaines tâches particulièrement critiques, on pourra supposer notamment une distribution détérministe : $Pr\left[d_i = D_i\right] = 1$. Pour d'autres, on pourra utiliser une distribution discrète. Par exemple, si l'on sait que l'exécution de la tâche i dépend d'un événement qui se produit avec la probabilité p, on pourra utiliser la distribution suivante : $Pr\left[d_i = \epsilon\right] = 1 - p$ et $Pr\left[d_i = D_i\right] = p$. On évite ainsi, contrairement aux approches actuelles pour l'ordonanncement des systèmes temps réel, de supposer que la tâche s'exécute systématiquement avec son temps maximal.

Pour un tel système, la première question qui se pose est celle de l'*ordonnançabilité* : existe-t-il une politique d'ordonnancement qui permette de respecter les contraintes $Pr\left[S_i > \delta_i\right] \leq \pi_i$, $i \in \mathcal{T}$? Réciproquement, étant donné un algorithme d'ordonnancement, les caractéristiques des tâches et la vitesse du processeur, quels sont les π_i qui sont réalisables ? Dans toute leur généralité, ces questions sont extrêmement complexes. C'est pourquoi, nous ne considérerons qu'un ensemble restreint d'algorithme d'ordonnancement tels que FCFS, EDF, PS, et priorités fixes.

Malgré sa simplicité apparente, ce modèle soulève des difficultés théoriques considérables. L'évaluation exacte de la probabilité qu'une tâche soit hors délai semble extrêmement difficile. On s'orientera donc vers d'une part des méthodes numériques, et d'autre part vers la recherche d'une borne supérieure sur cette probabilité. On pourra pour cela utiliser par exemple des analyses asymptotiques en régime de fort traffic comme proposé dans [28, 30, 29, 15, 27, 6], ou bien des techniques de comparaison stochastique comme proposé dans [32, 13, 31]. La qualité des approximations obtenues sera évaluée en utilisant des simulations événementielles.

2.2.2 Tâche B.2 : Évaluation de l'ordonnançabilité par méthode formelle

Un travail mené au sein de l'équipe OLC a permis de définir un langage, POLA [36], dédié à la spécification des systèmes de tâches temps réel et dont les modèles sont vérifiables par model checking. Afin de garantir ce caractère vérifiable, il a été indispensable de fournir une séman-

tique opérationnelle, en terme de réseaux de Petri temporels (RdPT). Ceux-ci se sont révélés trop limités pour modéliser certains comportements des ordonnanceurs et sont notoirement peu adaptés à une modélisation compositionnelle. Ces deux problèmes ont été résolus en étendant le formalisme des RdPT à l'aide de deux nouvelles relations, dites d'inhibition et de permission, permettant d'imposer une priorité entre évènements et de déplacer l'information temporelle des transitions dans le but d'utiliser l'opération de composition parallèle sans perturber le comportement temporel. L'explorateur d'espace d'états de la boîte à outils Tina a été étendu afin de permettre la vérification des descriptions Pola.

De plus, de nombreux travaux existent dans le domaine des réseaux de Petri stochastiques (voir chapitre 9 de [14]), dont certains ont été menés au sein du groupe OLC [19]. Continuez ces travaux en introduisant ces abstractions dans le cadre de l'analyse de l'ordonnancement des systémes temps réel semble une voix intéressante et pourrait conduire à de nouveaux résultats.

En menant cette recherche en parallèle des travaux sur l'approche par file d'attente, ce sera l'occasion de faire des croisements thématiques. La différence majeure entre ces deux approches réside dans la possibilité qu'offre les RdP de prendre en compte des dépendances et des caractéristiques comportementales. Cette capacité s'accompagne d'une complexité accrue, qui impose des méthodes numériques pour la vérification de l'ordonnaçanbilité. Par ailleurs nous étudierons l'apport des modèles de l'algèbre des dioïdes qui a permis récemment d'obtenir des résultats analytiques sur l'évaluation de performance en ordonnancement cyclique [20] et dans les systèmes temps réel périodiques [17].

2.2.3 Tâche B3 : Optimisation des offsets

Pour le système considéré au paragraphe 2.2.1, la probabilité qu'une tâche soit hors délai dépend du vecteur $\mathbf{t} = (t_i)_{i \in \mathcal{T}}$ des dates de première exécution (offset). Il est naturel pour le concepteur du système de chercher à optimiser son ordonnancement en jouant sur ces *offsets*. Un des problèmes d'optimisation que nous étudierons dans ce contexte peut être formulée de la façon suivante : $\min_{\mathbf{t}} \max_{i \in \mathcal{T}} Pr\,[S_i > \delta_i]$ sous les contraintes $Pr\,[S_i > \delta_i] \leq \pi_i$, $i \in \mathcal{T}$. Dans ce cas, on cherche un ordonnancement admissible minimisant la probabilité d'être hors délai pour la tâche la plus critique. On pourra aussi étudier un critère plus équitable, cherchant à réduire la probabilité que les tâches soient hors délai de manière uniforme : $\min_{\mathbf{t}} \alpha$ sous les contraintes $Pr\,[S_i > \delta_i] \leq \alpha\,\pi_i$, $i \in \mathcal{T}$.

2.2.4 Tâche B4 : Ordonnancement avec allocation de ressources

Dans de nombreuses applications, l'architecture matérielle est une architecture distribuée composées de plusieurs processeurs et d'un réseau de communication. La problématique consiste alors à ordonnancer l'exécution des tâches sur les processeurs, en tenant compte de nombreuses contraintes (exclusion entre les tâches pour raison de sécurité, contraintes de mémoire, latence de communication maximales, etc.).

Supposons ainsi que les tâches $i \in \mathcal{T}$ doivent être ordonnancées sur un ensemble $\mathcal{P}$ de processeurs. Une solution est décrite par l'ensemble $\mathcal{T}_p$ des tâches affectées à chaque processeur $p \in \mathcal{P}$, ainsi que par le vecteur $\mathbf{t}$ des offsets initiaux. Une formulation possible du problème est la suivante :

$$
\begin{array}{ll}
\min_{\mathbf{t}} & \alpha \\
\text{s.t.} & \\
& \cup_{p \in \mathcal{P}} \mathcal{T}_p = \mathcal{T} \\
& \mathcal{T}_p \cap \mathcal{T}_q = \emptyset \qquad\qquad p, q \in \mathcal{P} \\
& Pr\left[S_i > \delta_i\right] \leq \alpha\, \pi_i \quad i \in \mathcal{T}
\end{array}
$$

Bien entendu, la formulation ci-dessus n'est proposée qu'à titre illustratif. D'autres critères pourront être envisagées. De même, des contraintes sur le placement peuvent être ajoutées au problème pour le rendre plus réaliste. Néanmoins, même dans sa forme la plus simle, ce problème est extrêmenent difficile puisqu'il s'agit d'un problème combinatoire faisant intervenir des contraintes non-linéaires. Nous envisageons l'utilisation de métaheuristiques pour résoudre ce type de problèmes.

2.3 Reconfiguration

Un autre volet du projet concerne la recherche anticipée de multiples solutions d'ordonnancement cyclique afin d'autoriser une reconfiguration en présence d'aléas. Il s'agit en fait de prévoir à l'avance les changements (modifications de durées de tâches, arrivée de nouvelles tâches, pannes de ressources) qui peuvent rendre inapplicable un ordonnancement généré à l'avance.

En ordonnancement purement réactif, l'approche consiste à retrouver en temps réel un nouvel ordonnancement (cyclique) réalisable par des modifications locales de la solution courante. En termes d'optimisation combinatoire, cette modification de l'ordonnancement préétabli s'apparente à de la recherche locale car on cherche, pour des raisons d'efficacité et de stabilité (voir plus bas), une nouvelle solution dans le voisinage de la précédente. Dans les systèmes embarqués à temps critique, le temps alloué à la recherche d'une nouvelle solution acceptable en temps réel est très court, ce qui interdit l'utilisation d'approches sophistiquées de réordonnancement en temps réel. Dans ce cas, une approche intéressante consiste donc à anticiper les reconfigurations nécessaires pour un ensemble fini (mais potentiellement très grand) de scenarii d'aléas.

De telles approches ont été étudiées pour des problèmes de planification en intelligence artificielle [16], ainsi que pour des problèmes d'ordonnancement d'atelier et de projet [4, 25, 5], sous le terme d'ordonnancement conditionnel, proactif ou encore robuste. Plusieurs difficultés apparaissent pour la conception de telles méthodes, accentuées par le caractère cyclique de l'ordonnancement.

- Comment représenter les solutions alternatives et les associer à des classes d'aléas ? Une énumération explicite des solutions correspondant à chaque scénario peut poser des problèmes calculatoires si le nombre de scénarii est grand. Certaines approches en ordonnancement acyclique [5] proposent de représenter des familles de solutions (situées dans un voisinage l'une de l'autre) sous la forme de groupes de tâches permutables. Ce type d'approche n'a à notre connaissance jamais été abordé en ordonnancement cyclique.
- Comment assurer la stabilité du système ? En cas d'aléa, une reconfiguration qui change en profondeur l'ordonnancement prévu peut provoquer une certaine instabilité du système. Ainsi, il est nécessaire de considérer des contraintes ou des objectifs de stabilité. Cette stabilité nécessite une mesure de distance entre deux solutions, à déterminer en fonction du domaine d'application. Dans le cas où le problème d'ordonnancement comprend une fonction objectif comme la maximisation du débit, il devient alors nécessaire de réaliser un compromis entre la maximisation de la stabilité et la maximisation du débit.

Nous comptons apporter des réponses à chacune de ces questions pour les problèmes d'ordonnancement sous contraintes de ressources dans les systèmes embarqués critiques, ce qui constitue un défi assez important car ce problème n'a à notre connaissance jamais été abordé.

2.3.1 Tâche C1 : Reconfiguration sans contrainte de ressource

Nous pourrons aborder dans un premier temps les méthodes de reconfiguration dans des problèmes sans contrainte de ressource pour lesquels des algorithmes polynomiaux existent pour résoudre le problème sans scenario d'incertitudes. Nous étudierons ainsi la complexité de la reconfiguration pour diverses classes d'aléas.

2.3.2 Tâche C2 : Reconfiguration avec contraintes de ressource

Dans un second temps, nous aborderons les systèmes comprenant des ressources partagées, qui donne des problèmes d'ordonnancement NP-difficiles, même dans leur version sans scenario d'incertitudes. Ces systèmes étant fortement contraints, nous nous baserons sur une association des approches d'optimisation discrète robuste basées sur des scenarii ainsi que sur des méthodes de programmation par contraintes, ce qui constituera également un apport méthodologique.

3 Partenaires, personnes impliquées et rôles

Le projet implique trois groupes du LAAS (Mogisa, MRS et OLC) et prévoit une collaboration avec un chercheur de l'Université Technique d'Eidhoven (Pays-Bas).

3.1 Groupes de recherche impliqués dans le projet

3.1.1 Modélisation, Optimisation et Gestion Intégrée de Systèmes d'Activités

Le groupe MOGISA concentre ses activités de recherche sur les systèmes dans lesquels des activités doivent être réalisées à l'aide de ressources. L'exécution de ces activités est soumise à des contraintes liées à l'utilisation et la disponibilité des ressources, à la réalisation de projets ou de processus. Les problèmes posés relèvent de l'affectation et l'ordonnancement de tâches, le transport de biens ou de personnes, la planification des approvisionnements, de la production et de la distribution.

Un point fort du groupe est de développer des méthodologies originales de résolution de ces problèmes (1) basées sur les notions de flexibilité et de robustesse, (2) au carrefour d'expertises : optimisation combinatoire (Recherche Opérationnelle) et approche par contraintes (Intelligence Artificielle).

3.1.2 Modélisation et contrôle des Réseaux et Signaux

L'activité de recherche du groupe MRS concerne le développement de nouvelles approches méthodologiques et algorithmiques pour l'analyse, la conception et l'évaluation de performances de réseaux et systèmes de télécommunications, ainsi que le traitement avancé des signaux et systèmes.

Le groupe mène plusieurs projets de recherche sur la modélisation du trafic, l'optimisation, la planification et la commande des réseaux, les systèmes de calcul en grappes et en grille,

le filtrage non linéaire appliqué au traitement du signal, le filtrage de Volterra, la réalisation et l'identification des systèmes non linéaires par la méthode héréditaire et la représentation diffusive.

3.1.3 Outils Logiciels pour la Communication

Les travaux du groupe OLC s'intéressent aux systèmes de communication critiques à fortes contraintes temporelles et exigences de qualité de service et de sécurité.

Concevoir de tels systèmes amène à définir et réaliser un ensemble de composants logiciels caractérisés par des contraintes temporelles et/ou coopératives fortes. Les processus de conception utilisés s'appuient sur des bases formelles et couvrent différentes étapes de la spécification à l'implémentation en passant par la description de l'architecture et des comportements, leur vérification formelle, leur validation et leur évaluation expérimentales.

3.1.4 Stochastic Operations Research, UT Eindhoven

Le groupe "Stochastic Operations Research" du départemement Mathématiques et Informatique de l'Université Technique d'Einhdoven développe des modèles et méthodes mathématiques pour la conception, l'analyse, l'optimisation, et le contrôle des systèmes stochastiques. Du point de vue méthodologique, le groupe appartient aux communautés des probabilités appliquées et recherche opérationtionelle stochastique. Une grande partie des travaux est focalisé sur l'analyse des marches aléatoire et la théorie des files d'attente. Ces travaux s'inspirent notamment des applications dans les domaines de l'informatique, des télécommunications, et de la logistique.

3.2 Personnes impliqués et rôles

Le tableau 3.2 décrit les participants ainsi que leur implication et leur rôle dans le projet (les * indiquent les responsables du projet). Nous précisons ci-dessous pour chacun les compétences qui seront exploitées dans le cadre du projet :

- **C. Artigues** : optimisation combinatoire et ordonnancement robuste,
- **O. Brun** : modélisation stochastique et optimisation,
- **P.-E. Hladik** : ordonnancement temps réel et vérification formelle,
- **D. Le Botland** : vérification formelle et optimisation combinatoire,
- **B. Prabhu** : files d'attente et optimisation,
- **P. Esquirol** : programmation par contraintes et ordonnancement,
- **L. Houssin** : algèbre des dioïdes et ordonnancement,
- **M. Jonckheere** : files d'attente et comparaison stochastique.

4 Durée et phasage global

Le projet se déroulera sur 2 ans. La figure 3 décrit le phasage du projet. Nous prévoyons dans les 4 premiers mois (tâche A) de spécifier les cas d'études et de formaliser les contraintes à prendre en compte et les objectifs pour chacune des deux thématiques traitées. Pour ce qui concerne la thématique "intégration des incertitudes sur la demande", la première année sera consacrée aux aspects modélisation stochastique et vérification de l'ordonnançabilité (B1 et B2), tandis que la seconde portera plus sur les aspects optimisation de l'ordonnancement (B3 et B4). Pour la thématique "reconfiguration", la première année portera sur la reconfiguration sans contrainte de ressource (C1), ces contraintes étant introduites dans la deuxième année du projet (C2).

Nom	Fonction	Groupe	Tâches	Implication (%)
C. Artigues *	CR CNRS	LAAS/Mogisa	A, B3, B4, C1, C2	40
O. Brun *	CR CNRS	LAAS/MRS	A, B1, B2, B3, B4, C1, C2	40
P.-E. Hladik *	MC INSA	LAAS/OLC	A, B2, B3, B4, C1, C2	35
D. Le Botland	MC INSA	LAAS/OLC	A, B2, B3, B4, C1, C2	20
B. Prabhu	CR CNRS	LAAS/MRS	B1, B2, B3	20
P. Esquirol	MC INSA	LAAS/Mogisa	C1, C2	20
L. Houssin	MC UPS	LAAS/Mogisa	B2, C1, C2	20
M. Jonckheere	Assistant Professor	TU Eindhoven	B1, B3	20

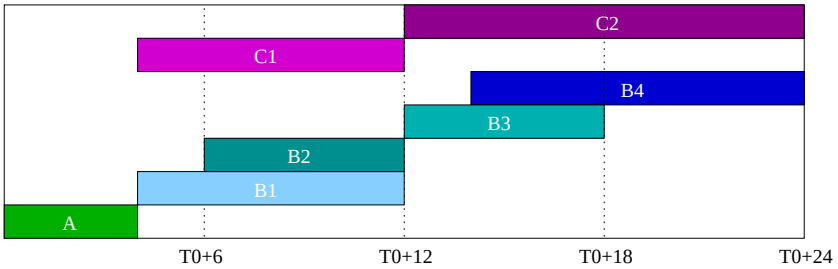

Fig. 3 – Phasage du projet.

5 Budget global et soutien demandé

Le coût global du projet est de 94328 euros. Le détail est le suivant :

- 18 mois de post-doc (69928 euros), qui se répartiront en un premier post-doc de 12 mois sur les tâches B1, B2 et B3, et un second post-doc de 6 mois sur la tâche C2.
- 3 stages de 6 mois par an (14400 euros) qui concerneront les tâches B1, B2 et C1 la première année, et les tâches B3, B4 et C2 la dernière année,
- Frais de déplacement pour la collaboration avec les Pays-Bas (2 séjours par an) et participation à des congrès (10000 euros).

Références

[1] Autosar homepage, 2008. `http://www.autosar.org/`.

[2] Arinc homepage, 2010. `http://www.arinc.com/`.

[3] S. Aalto, U. Ayesta, and R. Righter. On the gittins index in an m/g/1 queue. *Queueing Systems.*, 2009. To appear.

[4] M.A. Aloulou and M.C. Portmann. An efficient proactive-reactive scheduling approach to hedge against shop floor disturbances. In G. Kendall, E. K. Burke, S. Petrovic, and Michel Gendreau, editors, *Multidisciplinary Scheduling : Theory and Applications - 1st International Conference, MISTA '03*, pages 223–246. Springer, 2005.

[5] C. Artigues, J.-C. Billaut, and C. Esswein. Maximization of solution flexibility for robust shop scheduling. *European Journal of Operational Research*, 165 :314–328, 2005.

[6] R. O. Baldwin, N. J. Davis IV, J. E. Kobza, and S. F. Midkiff. Real-time queueing theory : A tutorial presentation with an admission control application. *Queueing Systems*, 35 :1–21, 2000.

[7] Pierre Bieber, Eric Noulard, Claire Pagetti, Thierry Planche, and François Vialar. Preliminary design of future reconfigurable ima platform. In *In 2nd Workshop on Adaptive and Reconfigurable Embedded Systems (APRES'09)*, 2009.

[8] P. Brucker and T. Kampmeyer. Tabu search algorithm for cyclic machine scheduling problems. *Journal of qscheduling*, 8(4) :303–322, 2005.

[9] Giorgio Buttazzo. *Hard Real-Time Computing Systems : Predictable Scheduling Algorithms and Applications*. Springer, 2005.

[10] C. Buyukkoc, P. Varaiya, and J. Walrand. The $c\mu$ rule revisited. *Advances in Applied Probability*, 17 :237–238, 1985.

[11] J. Carlier and P. Chretienne. *Timed Petri net schedules*, pages 62–84. Springer-Verlag, London, UK, 1988.

[12] P. Chrétienne. On graham's bound for cyclic scheduling. *Parallel computing*, 26 :1163–1174, 2000.

[13] J.L. Díaz, J.M. López, M. García, A.M. Campos, K. Kim, and L. Lo Bello. Pessimism in the stochastic analysis of real-time systems : concept and applications. In *25rd IEEE real-time systems symposium*, Lisboa, Portugal, December 2004.

[14] Michel Diaz, editor. *Les Réseaux de Petri : Modèles fondamentaux*. Hermès Science Publications, 2001.

[15] B. Doytchinov, J. Lehoczky, and S. Shreve. Real-time queues in heavy traffic with earliest-deadline-first queue discipline. *Ann. Appl. Probab.*, 11(2) :332–378, 2001.

[16] M. Drummond, J. Bresina, and K. Swanson. Contingent scheduling : Opportunities and issues. In *National Science Foundation workshop on Intelligent, Dynamic Scheduling for Manufacturing Systems*, 1993.

[17] S. Daniel F. Baccelli, B. Gaujal. Analysis of preemptive periodic real time systems using the (max, plus) algebra with applications in robotics. *IEEE Transactions on Control Systems Technology*, 10 :368–380, 2002.

[18] S. Faucou. *Description et Construction d'Architecture Opérationnelles Validées Temporellement*. PhD thesis, Université de Nantes, 2002.

[19] Laurent Gallon. *le modèle réseaux de Petri temporisés stochastique : extension et applications*. PhD thesis, Université Paul Sabatier de Toulouse, 1997.

[20] S. Gaubert and J. Mairesse. Modeling and analysis of timed petri nets using heaps of pieces. *IEEE Transactions on Automatic Control*, 44 :683–697, 1999.

[21] E. Gelenbe and I. Mitrani. *Analysis and Synthesis of Computer Systems.* Academic Press, 1980.

[22] J. C. Gittins. *Multi-Armed Bandit Allocation Indices*. Wiley, Chichester, 1989.

[23] C. Hanen and A. Munier. *Scheduling Theory and Its Applications*, chapter Cyclic Scheduling on Parallel Processors : An Overview. Wiley, 1995.

[24] C. Hanen and A. Munier. A study of the cyclic scheduling problem on parallel processors. *Discrete applied mathematics*, 57(2–3) :167–192, 1995.

[25] W. Herroelen and R. Leus. Project scheduling under uncertainty : Survey and research potentials. *European Journal of Operational Research*, 165 :289–306, 2005.

[26] H. P. Hillion and J. M. Proth. Performance evaluation of job-shop systems using timed event graphs. *IEEE Transaction on Automatic Control*, 34(1) :3–9, Jan. 1989.

[27] L. Kruk, J. Lehoczky, S. Shreve, and S. N. Yeung. Earliest-deadline-first service in heavy-traffic acyclic networks. *Ann. Appl. Probab.*, 14(3) :1306–1352, 2004.

[28] J. P. Lehoczky. Real-time queueing theory. In *RTSS '96 : Proceedings of the 17th IEEE Real-Time Systems Symposium*, page 186, Washington, DC, USA, 1996. IEEE Computer Society.

[29] J. P. Lehoczky. Real-time queueing network theory. In *RTSS '97 : Proceedings of the 18th IEEE Real-Time Systems Symposium*, page 58, Washington, DC, USA, 1997. IEEE Computer Society.

[30] J. P. Lehoczky. Using real-time queueing theory to control lateness in real-time systems. In *SIGMETRICS'97*, pages 158–168, Seattle, USA, 1997. ACM.

[31] H. Levy. Stochastic dominance and expected utility : survey and analysis. *Manag. Sci.*, 38(4) :555–593, 1992.

[32] J.M. López, J.L. Díaz, J. Entrialgo, and D. García. Stochastic analysis of real-time systems under preemptive priority-driven scheduling. *Real-Time Systems*, 40(2), 2008.

[33] P. Nain and D. Towsley. Optimal scheduling in a machine with stochastic varying processing rate. *IEEE Transactions on Automatic Control.*, 39 :1853–1855, 1994.

[34] M. Nuyens and A. Wierman. The foreground-background queue : a survey. *Performance Evaluation.*, 65 :286–307, 2008.

[35] G. J. Olsder, J. A. C. Resing, R. E. de Vries, M. S. Keane, and G. Hooghiemstra. Discrete event systems with stochastic processing times. *IEEE Transactions on Automatic Control*, pages 299–302, 1990.

[36] Florent Peres. *Réseaux de Petri temporels à inhibitions/permissions : Application à la modélisation et vérification de systèmes de tâches temps réel.* PhD thesis, Institut National des Sciences Appliquées Toulouse, 2010.

[37] R. Righter and J. G. Shanthikumar. Scheduling multiclass single server queueing systems to stochastically maximize the number of successfull departures. *Probability in the Engineering and Informationnal Sciences.*, 3 :323–333, 1989.

[38] L.E. Schrage. A proof of the optimality of the shortest remaining processing time discipline. *Operations Research*, 16 :687–690, 1968.

[39] L.E. Schrage and L. W. Miller. The queue m/g/1 with the shortest remaining processing time discipline. *Operations Research*, 14 :670–684, 1966.

[40] Kenneth Seeling. Reconfiguration in an integrated avionics design. In *in 15th AIAA-IEEE Digital Avionics Systems Conference*, 1996.

[41] D. R. Smith. A new proof of the optimality of the shortest remaining processing time discipline. *Operations Research*, 26 :197–199, 1978.