VACSIM

Validation de la commande des systèmes critiques par couplage simulation et méthodes d'analyse formelles

Tâche 2

Validation par simulation de partie opérative

Livrable L2.1

Spécification et validation, sur études de cas, d'une méthode d'utilisation des simulateurs de parties opératives : objectifs, propriétés de sûreté de fonctionnement, taux de couverture et limitations de la simulation

Version B

Appel :	**PROGRAMME INGENIERIE NUMERIQUE & SECURITE 2011**
Numéro d'agrément :	A**NR-11-INSE-004**
Thématique:	**Systèmes embarqués et ingénierie du logiciel**
Objectif:	**Validation par simulation de partie opérative**
Date de démarrage du projet:	**01.10.2011**
Durée :	**42 mois**

Synthèse

Le projet **VACSIM** (Validation de la commande des systèmes critiques par couplage simulation et méthodes d'analyse formelle), référencé ANR-11-INSE-004, étudie les avantages respectifs des techniques de simulation, en incluant des modèles des processus commandés, et des méthodes d'analyse formelles, pour la validation de la commande des systèmes critiques. Ce projet est structuré en 6 tâches.

La tâche 2 « Validation par simulation de partie opérative » vise à proposer une méthode d'utilisation automatisée d'un simulateur de partie opérative, afin d'augmenter l'efficacité de ce type de technique de validation.

Pour les systèmes de contrôle-commande complexes en effet, une pratique industrielle courante consiste à utiliser un simulateur de la partie contrôlée, aussi dénommée partie opérative, bouclé aux spécifications, puis à la réalisation, du système de contrôle-commande. Malheureusement, les cas connus de ce type de techniques correspondent à des utilisations manuelles du simulateur, sans méthode pour son utilisation, c'est-à-dire sans démarche guidant l'introduction d'états et de défauts sur le modèle de partie simulée pour la sollicitation de la partie commande, ni d'objectifs clairs de taux de couverture des tests de l'ensemble. Les démarches d'utilisation des simulateurs ne sont pas complètement formalisées. Cette difficulté scientifique et technique conduit concrètement à l'impossibilité d'automatiser ce type de test, du fait de cette absence de formalisation. Ceci peut aboutir à un coût prohibitif et à des efforts démesurés quand une exigence d'un grand nombre de cas de tests est posée, typiquement pour le test statistique d'un système critique. Le nouveau projet VACSIM est l'occasion d'examiner les possibilités de résoudre cette difficulté. L'objectif est ici de développer un environnement de simulation de systèmes critiques, intégrant l'analyse de propriétés de sûreté de fonctionnement, qui puisse automatiser une utilisation méthodique du simulateur qui aura été définie dans le projet, sur la base d'objectifs et de taux de couverture de test rationnellement définis.

Ce livrable L2.1 du projet VACSIM décrit une spécification et propose une validation sur études de cas d'une méthode d'utilisation des simulateurs de parties opératives : objectifs, propriétés de sûreté de fonctionnement, taux de couverture et limitations de la simulation.

Ce livrable L2.1 a été initialement rédigé par la société EDF R&D, puis complété par la société Dassault Systèmes, tenant compte du retour d'expérience dans l'utilisation des simulateurs que la société fournit, et par les partenaires académiques du projet, tenant compte de l'application possible des formalismes, modèles et méthodes qu'ils approfondissent dans les autres tâches du projet.

Sommaire

1. .Introduction

1.1. Cadre de l'étude

L'objectif principal du projet VACSIM est de tirer profit des avantages respectifs des techniques de simulation, en incluant des modèles des processus commandés, et des méthodes d'analyse formelles, pour la validation de la commande des systèmes critiques. Le projet s'intéresse au couplage des approches formelles de test et de vérification avec des approches de simulation, ces deux approches ayant des avantages complémentaires, notamment en termes de capacité à passer à l'échelle et de maîtrise du taux de couverture de l'analyse. Le projet vise à développer des contributions de nature à la fois méthodologique (définition de nouveaux modes d'utilisation des simulateurs, règles de couplage simulation / méthodes formelles) et formelle (adaptation, extension ou création de méthodes formelles) qui permettront la réalisation d'un démonstrateur, sur la base des outils d'ingénierie numérique et de simulation ControlBuild Validation et Dymola de la société Dassault Systèmes, illustrant, sur la base d'études de cas industriels, les bénéfices du couplage. Le but ultime du projet est de proposer un continuum de validation durant le cycle de vie de la commande des systèmes critiques basé sur des environnements d'ingénierie numérique. L'expertise des dysfonctionnements en phase d'exploitation devrait aussi profiter de l'utilisation automatisée des simulateurs pour l'étude des défauts et de leurs conséquences qui est proposée dans ce rapport.

1.2. Etat de l'art sur la simulation de partie opérative et définition du besoin

Ce chapitre rappelle l'état de l'art réalisé lors de la préparation du programme VACSIM.

Pour la validation des systèmes de commande, une pratique industrielle courante consiste à utiliser un simulateur de la partie opérative connecté aux spécifications, puis à la réalisation du système de commande. L'offre en outils de simulation de partie opérative comprend, par exemple :

- Les outils MATLAB, pour le calcul, et Simulink, pour la représentation des fonctions mathématiques et des systèmes par diagramme en blocs, de la société Mathworks, qui peuvent être utilisés pour modéliser la partie commande, les instruments et la partie commandée (partie opérative) afin de valider un fonctionnement d'ensemble.
- Pour des modélisations plus complexes, qui obligent à intégrer des équations explicites (équations différentielles, polynômes), l'utilisation de langages plus élaborés comme Modelica permet une simulation multi-domaines temps réel de systèmes complexes. Ces langages peuvent être intégrés dans des environnements de simulation libres ou commerciaux comme Dymola de la société Dassault Systèmes.
- En plus de ces outils généralistes, d'autres outils sont optimisés pour une activité donnée en contrôle industriel ou dans un métier particulier. Par exemple l'outil ControlBuild Validation de Geensoft / Dassault Systèmes, qui peut être utilisé pour simuler le comportement des équipements électromécaniques des instruments, le comportement physique d'une installation et représenter des pupitres de conduite, en offrant la possibilité d'injecter des défauts pour une validation en usine du contrôle et une formation des opérateurs.
- Des simulateurs de formation pleine échelle d'installations de grande taille, telles que des centrales électriques, existent, par exemple les simulateurs de la société CORYS pour le transport et l'énergie.
- Pour le besoin de leurs études lors de programmes critiques, certains industriels comme EDF ont pu développer sur mesure des outils métier qui leur sont propres.
- De grands fournisseurs comme ALSTOM, en particulier dans l'énergie, ont aussi développé une offre d'outils pour simuler l'installation, la partie contrôle et les interfaces de conduite, utilisable pour une validation en usine de systèmes de contrôle.

Si la modélisation du comportement physique des systèmes est de plus en plus utilisée dans l'industrie lors de la conception de nouveaux produits, le fonctionnement de ces systèmes est en général soumis à un certain nombre de conditions, qu'il s'agisse de contraintes physiques, d'exigences ou d'hypothèses, regroupées sous le terme de propriétés. Ces propriétés ont fait l'objet de travaux dans le cadre du projet ITEA2 EuroSysLib, auquel participaient EDF et Dassault Systèmes, qui s'est terminé en 2010 et avait pour objectif de développer le langage Modelica, langage de modélisation physique de plus en plus utilisé dans l'industrie, et ses bibliothèques couvrant différents domaines. Il est donc possible, lors de l'utilisation de simulateurs couplés à une partie commande critique, de représenter les propriétés attendues d'un système, en particulier de sûreté de fonctionnement et de performance, et de vérifier lors des simulations que ces propriétés sont effectivement satisfaites.

Une approche consistant à associer un modèle de comportement Modelica à un modèle de propriétés exprimé dans un autre langage a aussi été étudiée dans le cadre du projet ITEA2 OpenProd, actuellement en cours d'étude avec le développement du profil UML ModelicaML.

Par contre, les cas connus de ce type de techniques correspondent à des utilisations manuelles du simulateur, sans méthode pour son utilisation, c'est-à-dire sans démarche guidant l'introduction d'états et de défauts sur le modèle de partie simulée pour la sollicitation de la partie contrôle, ni d'objectifs clairs de taux de couverture des tests de l'ensemble. Il n'existe pas de méthode reconnue définissant les utilisations de ces simulateurs pour la validation de la commande des systèmes critiques : choix des défauts sur la partie installation, sur la partie instruments, sur la partie contrôle-commande, critères de couverture de ces différentes parties lors de leur simulation.

Les verrous techniques auxquels s'attaque ce livrable sont relatifs aux méthodologies d'utilisation des simulateurs de processus et au couplage entre méthodes de simulation, vérification et test, en s'intéressant à la problématique de modélisation multi-échelles. En complément et pour la définition d'un environnement automatisé d'utilisation des simulateurs, plusieurs verrous scientifiques sont à lever afin d'améliorer le niveau de maturité des méthodes d'analyse formelle qui pourraient être intégrées à cet environnement ; ils concernent la décidabilité et la complexité du model-checking de certaines classes d'automates finis temporisés, la résolution de systèmes de contraintes non linéaires sur les flottants, la détermination de taux de couverture par identification de systèmes à événements discrets, et le développement de stratégies permettant le passage à l'échelle de ces méthodes.

1.3. Objectif du document

L'objectif de la tâche 2 du projet VACSIM est de proposer une méthode d'utilisation automatisée d'un simulateur de partie opérative, afin d'augmenter l'efficacité de ce type de technique de validation.

Pour les systèmes de commande complexes en effet, une pratique industrielle courante consiste à utiliser un simulateur de la partie contrôlée, aussi dénommée partie opérative, bouclé aux spécifications, puis à la réalisation, du système de contrôle-commande. Malheureusement, les cas connus de ce type de techniques correspondent à des utilisations manuelles du simulateur, sans méthode pour son utilisation, c'est à dire sans démarche guidant l'introduction d'états et de défauts sur le modèle de partie simulée pour la sollicitation de la partie commande, ni d'objectifs clairs de taux de couverture des tests de l'ensemble. Les démarches d'utilisation des simulateurs ne sont pas complètement formalisées. Cette difficulté scientifique et technique conduit concrètement à l'impossibilité d'automatiser ce type de test, du fait de cette absence de formalisation. Ceci peut aboutir à un coût prohibitif et à des efforts démesurés quand une exigence d'un grand nombre de cas de tests est posée, typiquement pour le test statistique d'un système critique. Le nouveau projet VACSIM est l'occasion d'examiner les possibilités de résoudre cette difficulté. L'objectif est ici de développer un environnement de simulation de systèmes critiques, intégrant l'analyse de propriétés de sûreté de fonctionnement, qui puisse automatiser une utilisation méthodique du simulateur qui aura été définie dans le projet, sur la base d'objectifs et de taux de couverture de test rationnellement définis.

L'objectif de ce livrable L2.1 du projet VACSIM est de décrire une spécification et de proposer une validation sur études de cas d'une méthode d'utilisation des simulateurs de parties opératives : objectifs, propriétés de sûreté de fonctionnement, taux de couverture et limitations de la simulation.

1.4. Cas d'utilisation des tests, de la simulation et des preuves formelles

Suivant les secteurs industriels et leur criticité, des preuves de la qualité des systèmes peuvent être obtenues par des tests, des simulations et des vérifications pouvant utiliser des preuves formelles.

Pour des protections critiques, on évite le plus souvent d'augmenter le risque en introduisant des objets complexes et la simplicité est recherchée. Le cas général consiste à accorder plus d'importance au dépassement d'une limite de fonctionnement autorisé, activant une fonction logique de protection, qu'au maintien d'un point de fonctionnement dans une zone de fonctionnement optimale, ce qui utilise en général des régulations. Les protections les plus critiques sont de ce fait le plus souvent des protections logiques simples. Elles interviennent ou non suivant que le point de fonctionnement est dans une zone de fonctionnement autorisée ou non[1]. C'est donc la reconnaissance d'une situation de fonctionnement qui est utilisée en entrée d'une chaîne de sécurité critique plus que l'événement d'une transition entre deux situations. Ces protections sont généralement testées en boucle ouverte en simulant en entrée une situation pour juger de la pertinence d'une sollicitation ou d'une absence de sollicitation d'une fonction de sécurité. Ces tests peuvent être d'abord réalisés sur des spécifications exécutables des protections, afin de valider les cas de sollicitation, puis sur le système réel, dans un test de conformité (et même sur le système réel sur site une fois l'environnement de programmation désactivé pour les systèmes les plus importants en nucléaire).

En aéronautique comme en nucléaire, certaines parties de systèmes de contrôle ne peuvent être considérées comme complètement qualifiées une fois testées, même en utilisant un grand nombre de simulations représentatives, du fait d'un grand nombre d'états et de configurations possibles dans leur utilisation. C'est par exemple le cas de certains bus de communication un peu critiques, pour lesquels des tests ou des simulations permettent difficilement d'être sûr d'avoir reproduit le cas le plus pénalisant dans la transmission de données. Pour ces réseaux, il peut être demandé d'apporter des preuves de prédictibilité des résultats, au sens du déterminisme du comportement d'un modèle du système en relation avec son environnement en nucléaire [5], ou calcul du pire cas de communication pour la qualification des réseaux en aéronautique. Ces preuves formelles sont réalisées sur des modèles du système de communication, par exemple avec des techniques de type « Network Calculus ».

A l'inverse, en nucléaire, une preuve formelle de propriétés sur un modèle de l'installation ne saurait suffire à prouver la sûreté de la centrale dans le cas de situations accidentelles prises en compte dans sa conception et son dimensionnement. Une action de protection est prévue pour tout incident et accident envisagé du Rapport de Sûreté (RdS) de la centrale et des codes de calcul qualifiés sont utilisés pour analyser par simulation, de façon réaliste, chaque transitoire accidentel. La simulation est ainsi utilisée pour sanctionner une approche déterministe de la conception du système de protection.

1.5. Méthodologies dans l'emploi de codes de calcul et de simulateurs pour le test

Nous disposons actuellement principalement de **deux méthodologies différentes** de l'utilisation de codes de calcul et d'outils de simulation de processus industriels critiques :

- La simulation pour la génération d'un transitoire réaliste dans un scénario de test en boucle ouverte d'un système de protection ; (cf. Figure 1)
- La simulation conjointe en boucle fermée de la partie contrôle et de la partie mécanique contrôlée dans la production d'un transitoire de test devant vérifier des propriétés sur ces deux parties. (cf. Figure 3)

[1] Il s'agit d'une simplification dans l'exposé, car le calcul d'un seuil de protection doit tenir compte de l'imprécision de l'instrumentation, de la dérive éventuelle entre deux calibrages, de la dynamique des transitoires considérés et du temps de réponse des chaînes de sécurité.

Dans le **premier cas**, un code de calcul, ou plus généralement un outil de simulation de la partie à contrôler, est utilisé pour obtenir des valeurs physiques représentatives d'un transitoire nécessitant une action de protection d'un système de sécurité. Ces scénarios d'évolutions de paramètres sont ensuite utilisés en entrée d'un système sous test (qui peut être représenté avec plusieurs niveaux de détail, depuis les spécifications exécutables du système de protection jusqu'au système réel). L'évolution réaliste des variables du procédé lors du transitoire simulé en entrée du système sous test permet de contrôler sa non-réaction tant qu'il n'a pas à réagir puis le déclenchement d'ordres de protection. Pour éviter une analyse purement manuelle des résultats du test, on utilise un oracle censé représenter le résultat attendu du système de protection et l'on calcule les différences entre les réactions de l'oracle et du système à tester.

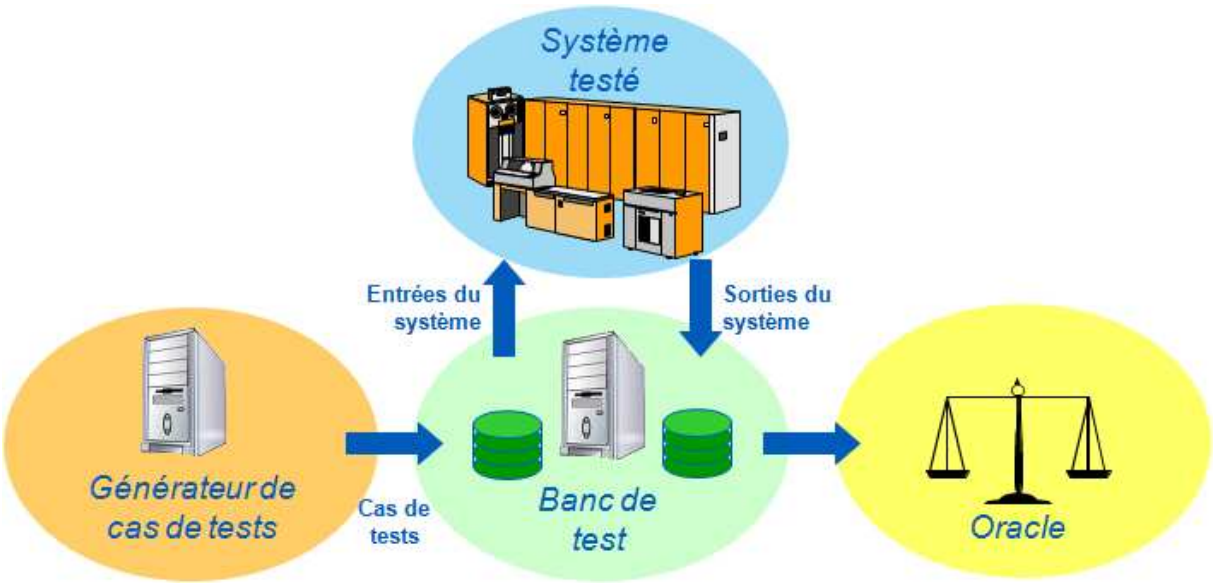


Figure 1 : Méthodologie de réalisation de tests « boucle ouverte »

Une variante de l'approche (cf. Figure 2) consiste à ne pas utiliser directement le système à tester mais une simulation de ce système en parallélisant éventuellement l'exécution de l'oracle et du système à tester. Ceci présente l'avantage de pouvoir exécuter en parallèle de nombreux tests et de pouvoir valider ou au moins vérifier la qualité du système plus tôt dans le développement, sans avoir à attendre la réalisation complète du système.

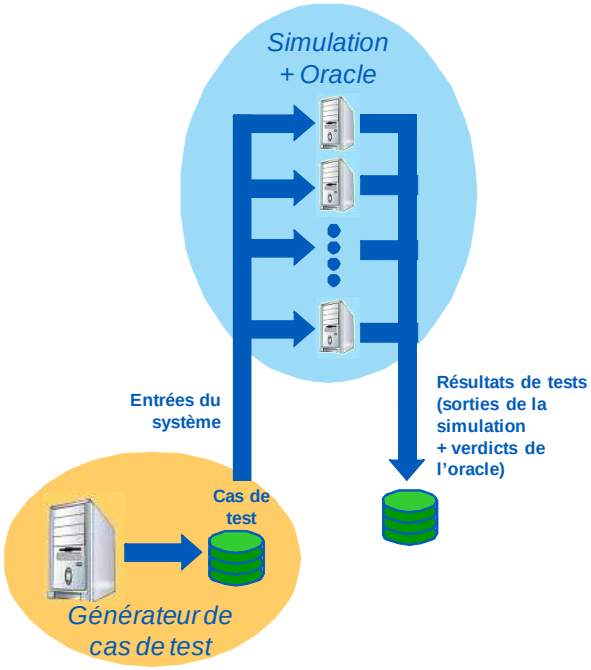


Figure 2 : Parallélisation des simulations « boucle ouverte »

La première limite de cette première approche est l'obligation de disposer d'un oracle de qualité, ce qui conduit finalement à recréer des spécifications de l'attendu fonctionnel du système à tester. Un problème de spécification du système à tester peut être reproduit dans les spécifications de l'oracle, ce qui conduit à un défaut de cause commune de spécification. Pour des systèmes critiques, on peut vouloir utiliser des équipes indépendantes pour la spécification du système à tester et de son oracle. En résultat, les deux spécifications peuvent comporter de subtiles différences dans l'expression de l'attendu, qui conduiront à des discordances dans l'exécution du test. Ces discordances devront faire l'objet d'analyses manuelles. Par ailleurs, l'oracle et le système à tester ne peuvent réagir à un moment strictement identique. Il faut donc aussi savoir gérer les incertitudes dans les évolutions des paramètres d'entrée, pour juger de la sanction à apporter au test, sous peine d'augmenter le nombre de fausses discordances de façon drastique.

Surtout, cette approche ne permet pas de simuler correctement le transitoire résultant de l'activation du système de protection testé pour vérifier que l'ensemble constitué de la mécanique et du système de protection vérifie des propriétés de sûreté et reste dans des limites de fonctionnement acceptables.

Cette première utilisation présente donc des limites et est complexe à mettre en œuvre. Cette complexité augmente avec la complexité des systèmes contrôlés et des systèmes de contrôle, ce qui peut obliger à **choisir la seconde méthodologie** d'utilisation des codes de calcul et des simulateurs pour le test. Ainsi, dans son utilisation des outils de simulation fournis par Dassault Systèmes, la société productrice d'automobile BMW, comme la société Alstom pour les Trains à Grande Vitesse, mettent en avant l'intérêt de simuler de façon conjointe les parties mécaniques et de contrôle-commande, chaque partie étant devenue trop complexe pour pouvoir être testée séparément.

La seconde approche consiste donc à simuler conjointement la partie contrôle et la partie à contrôler. On parle alors de co-simulation. Le scénario de test consiste à initialiser dans un état donné le simulateur du processus contrôlé – la partie mécanique – et le système de contrôle critique testé. Cet état peut comprendre des défauts sur chacune de ces parties, qui conduiront à la co-simulation globale d'un transitoire. Dans cette co-simulation, le couplage entre la partie contrôle et la partie opérative est obtenue par le bouclage des sorties du système de contrôle testé sur le simulateur de processus, qui lui fournit des valeurs d'entrée. Dans une utilisation classique, on initialise le simulateur et le système à tester dans un état stable, puis le scénario de test injecte un défaut sur la partie mécanique qui doit conduire à l'activation d'une fonction du système de contrôle à tester et on enregistre l'évolution de l'ensemble des variables. L'oracle consiste alors à vérifier que des propriétés sont vérifiées. Il peut s'agir de propriétés de performance sur le système de contrôle à tester, par

exemple des charges des réseaux de communication, mais surtout, d'un point de vue fonctionnel[2], des propriétés de sûreté de fonctionnement sur le processus contrôlé, comme le non-dépassement de limites de contraintes mécaniques, de températures, de pression et autres limites autorisées de fonctionnement et le respect d'exigences de conception ou d'exploitation, comme l'utilisation d'une puissance inférieure à la puissance maximale disponible ou le respect de gradients de refroidissement par exemple.

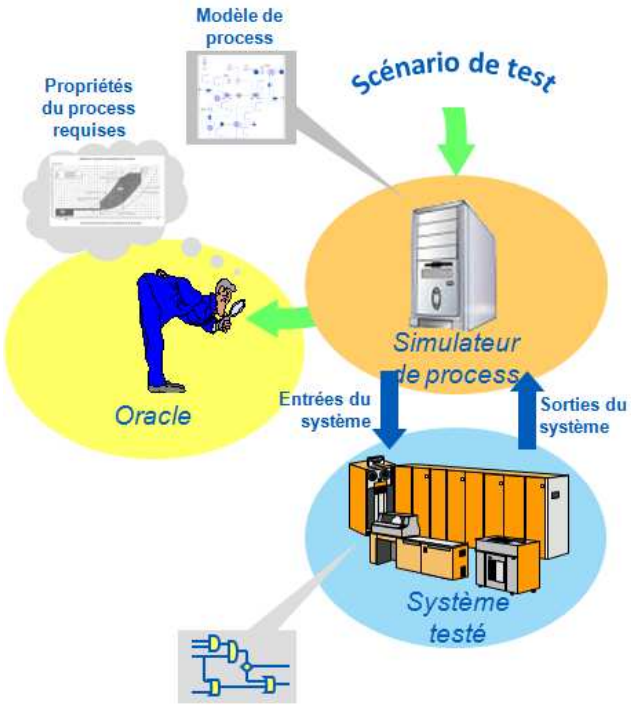


Figure 3 : Méthodologie de réalisation de tests « boucle fermée »

C'est cette méthodologie d'utilisation des simulateurs pour la validation de la commande de systèmes critiques qui est préconisée dans le projet VACSIM et était déjà sous-entendue dans les discussions préparatoires à ce programme. On peut d'ailleurs noter que, de façon générale, l'utilisation du système bouclé (commande + processus) en lieu et place de la commande considérée isolément (boucle ouverte), est une solution efficace reconnue pour réduire la taille de l'espace d'état, lors d'une analyse, notamment en vérification formelle [9][10].

2. Méthodes d'utilisation des simulateurs et verrous scientifiques et techniques du projet VACSIM

2.1. Cas d'étude et première proposition d'une démarche

Dans ce chapitre et dans la suite du document, on entend par « partie opérative » a minima l'interface avec le procédé. On entend par « simulation de partie opérative » la simulation des capteurs et actionneurs et, par extension et en raccourci, la simulation du système physique à contrôler et à commander, soit la simulation des capteurs et actionneurs et de la partie mécanique.

Si on pense rapidement à valider un système critique en activant ses fonctions de sécurité par une simulation de défauts sur la partie contrôlée afin de prouver l'efficacité des fonctions de protection, il est curieusement plus rare de valider un bon fonctionnement normal. Conformément aux bonnes

[2] En général, on réalise un système de contrôle pour un processus et non l'inverse.

pratiques de l'industrie nucléaire, il est conseillé de valider que l'utilisation normale du procédé ne conduit pas à un déclenchement intempestif des protections du système critique de contrôle. Les travaux du projet devront aboutir à une utilisation des simulateurs pour, dans un ordre croissant de complexité :

- Valider le bon fonctionnement normal (démarrage, décollage, arrêt, atterrissage…) et les grands transitoires normaux d'exploitation (îlotage, changement de régime de puissance…) en vérifiant le non-déclenchement intempestif des protections du système critique de contrôle ;
- Valider le respect de limites de fonctionnement et de propriétés de sûreté en cas d'incidents sollicitant les fonctions de sécurité du système critique de contrôle.

Pour ce second point, on peut commencer par utiliser les simulateurs en injectant les défauts prévus à la conception de la partie mécanique contrôlée (fuite, perte de débit, bas niveau, ouverture ou blocage en position de vannes, pannes de pompes ou de turbines, perte d'alimentation électrique de puissance...) qui font normalement l'objet d'alarmes et/ou d'actions automatiques. Dans un second temps, on peut simuler des pertes d'instruments, capteurs ou actionneurs, de la partie opérative (perte d'alimentation électrique d'un capteur ou défaillance de la chaîne d'instrumentation, perte d'air comprimé pour des vannes pneumatiques, de tension de commande pour des disjoncteurs et des contacteurs…). Enfin, on peut simuler des pannes matérielles de la partie automatisme du système critique de contrôle (perte d'une liaison de communication, d'une alimentation électrique d'un module de contrôle[3]…).

Sans démarche méthodique, une installation industrielle ou un système un peu complexe peuvent conduire à envisager d'injecter des milliers de défauts sur la partie mécanique et autant sur le système critique de contrôle. Si l'on envisage la concomitance d'une panne mécanique et d'une panne du système de contrôle, l'espace des possibles conduit rapidement à un nombre trop important de situations potentiellement simulables. La démarche proposée pour éviter ce problème combinatoire consiste à ne simuler que les cas suivants :

- Le fonctionnement normal sans incident sur la partie mécanique et sans défaut sur la partie contrôle : vérification de l'absence d'ordres intempestifs ;
- Le fonctionnement normal de la partie mécanique en cas de défaut sur la partie contrôle (instrumentation, automatismes de mise en service, de protection ou de régulation ou interface du système de conduite et de supervision) : vérification du fonctionnement des mécanismes prévus de reprise en secours et de gestion des pannes de la partie contrôle (gel ou passage en position de repli de sorties, redondance active ou passive de serveurs, alimentation électrique redondante d'unités de traitement, anneau ou redondance des bus de communication, signalisation et passage à un pupitre de conduite de secours opérationnel) ;
- Le fonctionnement en cas d'incident sur la partie mécanique prévu à la conception sans défaut sur la partie contrôle : vérification de l'efficacité des fonctions de sécurité ;
- Le fonctionnement en cas d'incident sur la partie mécanique prévu à la conception en présence d'un défaut sur la partie contrôle qui est sollicitée lors du transitoire : vérification du maintien des exigences globales de performance et de sûreté lors de la sollicitation des moyens prévus pour la redondance, pour la reprise en secours et pour les positions de repli en cas de panne du contrôle-commande ainsi que pour la diversification fonctionnelle du contrôle-commande et des systèmes mécaniques.

[3] Des pannes du contrôle-commande plus complexes comme la génération d'ordre intempestif, la perte d'événement ou le dépassement d'une capacité de calcul pourraient aussi être envisagées, mais l'automatisation de simulations incluant des pertes de communication et des pertes de sources électriques permettrait déjà de solliciter les principaux mécanismes de redondance et de basculement prévus de l'architecture.

2.2. Parties modélisées et niveau de détail des modèles en simulation

2.2.1. Parties modélisées suivant quelques pratiques de simulation

La conduite d'une installation ou, plus généralement, l'utilisation d'un système via un système de contrôle, amène classiquement à représenter plusieurs parties :

- La partie mécanique (le processus) ;
- Les actionneurs et capteurs réalisant l'interface avec le procédé (appelé niveau 0 d'une architecture de contrôle-commande) ;
- Les automatismes réalisant des fonctions de régulation, de protection, de mise en service et à l'arrêt, de verrouillage, de basculement normal/secours et d'élaboration d'alarme (niveau 1) ;
- Les IHM permettant la conduite et la supervision (niveau 2) ;
- Les opérateurs, pilotes ou utilisateurs qui utilisent des procédures de conduite ou d'exploitation, des manuels d'utilisation.

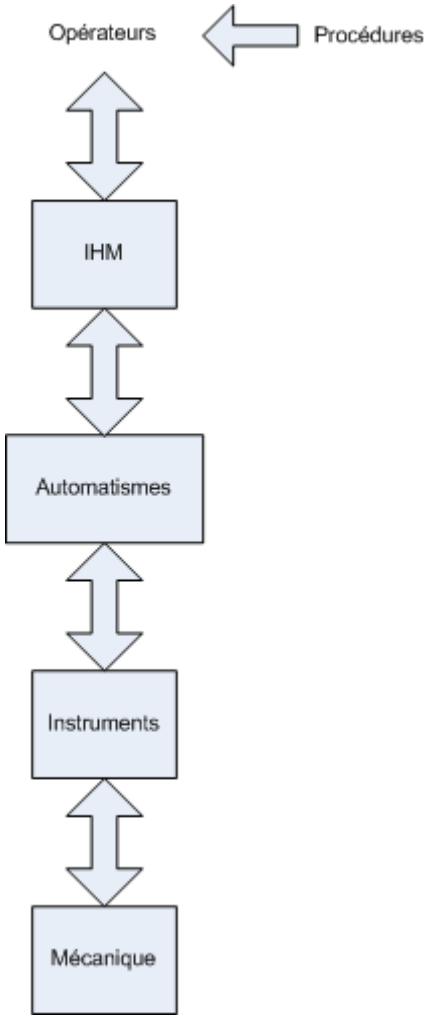


Figure 4 : Les parties à simuler

Les fonctions de configuration du contrôle-commande et de la gestion technique, qui correspondent au niveau 3, ne sont pas détaillées dans ce chapitre, mais elles seront à considérer pour la définition fonctionnelle du « superviseur de la simulation » qui est introduit dans les chapitres suivants.

En pratique, toutes les parties ne sont pas systématiquement représentées pour une simulation de l'ensemble de l'installation. Le périmètre de la représentation peut varier en fonction de ce qu'on cherche à valider.

Dans le cas d'une validation d'une protection critique, la partie mécanique peut être représentée de façon très rigoureuse en utilisant des codes de calcul qualifiés alors que seuls les instruments et les protections sollicités peuvent être simulés. Les temps de réponse et les précisions des chaînes de sécurité font l'objet d'exigences quantifiées, qui donnent lieu respectivement à des temporisations et des décalages des seuils des chaînes de protection dans les simulations du transitoire résultant de l'incident étudié. Des aggravants sont aussi introduits pour tenir compte du pire cas de fonctionnement des parties non modélisées.

Dans le développement d'un système de contrôle-commande pour l'énergie, une pratique courante consiste à reboucler les sorties du système de contrôle sur ses entrées pour simuler rapidement les spécifications de fonctions de contrôle simples ou solliciter les blocs de traitement du système lors de sa programmation (l'ordre d'ouverture d'une vanne est rebouclé sur le compte-rendu de vanne ouverte). La plupart des environnements de développement des Systèmes Numériques de Contrôle-Commande (SNCC) offrent cette première possibilité de « simulation ». Dans cette représentation très simple de la partie mécanique et/ou des instruments, il est aussi possible d'introduire des temporisations pour représenter un temps de manœuvre d'un actionneur commandé ou de mise en service d'une pompe. Les temporisations peuvent aussi être utilisées pour simuler grossièrement certaines évolutions de paramètres continus pour des phénomènes simples. Par exemple, l'ouverture d'une vanne d'arrivée peut activer une temporisation qui « simulera » l'évolution continue du niveau d'une bâche pour activer une alarme ou un ordre automatique de fermeture de la vanne à un seuil de niveau donné.

Pour des lois de commande un peu plus complexes que des protections logiques, on peut vouloir représenter la partie mécanique de façon moins grossière. L'attendu fonctionnel de traitements analogiques tels que des régulations est moins simple à exprimer que pour des protections logiques simples tel qu'évoquées plus haut (au chapitre 1.4). Pour des régulations, plusieurs critères sont à utiliser pour juger de leurs performances, dont les temps de réponse, l'atteinte de la consigne et la stabilité. Une pratique industrielle courante consiste à représenter la partie mécanique par des fonctions de transfert et à simuler le fonctionnement d'ensemble. La réponse à un changement de mode de fonctionnement de la mécanique et de sa régulation est le plus souvent jugée à dire d'expert. Certaines industries de certains pays codifient toutefois dans des normes et recommandations techniques la réponse standard à des variations types d'entrées de régulateurs connus. Quand ce n'est plus la spécification qui est testée par simulation mais le système réel qui a été développé, l'usage est de tester les régulations en utilisant les interfaces homme / machine (IHM) du système, en vérifiant ainsi certaines fonctions de base comme les affichages de courbes ou les commandes de basculement entre modes automatique et manuel, mais aussi les reports d'archivage et l'élaboration d'alarmes par exemple. Pour une représentation plus fine des parties mécaniques régulées, certains environnements de test utilisent des blocs de fonctions mathématiques ou des équations physiques.

EDF a développé un environnement pour valider des consignes de conduite de la CIA (conduite incidentelle accidentelle) par leur utilisation automatisée avec un simulateur de procédé (outil Scoop et animateur Antares d'EDF SEPTEN). La consigne manuelle de conduite à tester est d'abord transformée en module exécutable. L'environnement permet de coupler ce module à un simulateur pleine échelle d'une centrale nucléaire pour réaliser une conduite virtuelle du procédé. La définition d'un scénario de test comprend le choix d'une situation initiale à partir de laquelle on introduit l'incident devant conduire à l'utilisation de la consigne, plus des événements devant se produire à échéance définie lors du transitoire. Le nombre important de scénarios réalisables permet d'envisager une couverture important du parcours dans les consignes (les branches d'une consigne). Par exemple, si la procédure conduit à utiliser un système redondé de refroidissement, la consigne comprendra un pas test orientant vers l'utilisation d'une file principale de refroidissement, si elle est disponible, ou vers l'utilisation d'une file de refroidissement de secours. Dans le scénario de test de la

consigne, l'introduction d'une panne sur une pompe de la file principale de refroidissement conduira à basculer sur la file de refroidissement de secours dans l'utilisation automatisée et en boucle de la consigne. Les variables du procédé à analyser sont archivées lors du transitoire afin de permettre leur restitution à l'issue de l'exercice.

L'automatisation de l'exécution de la consigne permet de définir des critères objectifs de parcours de la consigne (couverture), d'en rendre compte et de pouvoir réaliser automatiquement un grand nombre de simulations. Ceci facilite les études de sensibilité, qui examinent la robustesse de la conduite prévue à une petite variation de paramètres et de conditions initiales (on n'étudie plus un transitoire à partir d'un point de fonctionnement mais plusieurs transitoires à partir d'un nuage de points, d'où de nombreuses simulations). Les avantages les plus significatifs de l'approche sont liés à la reproductibilité du test, à l'étude de la robustesse aux défaillances, au calcul automatisé du délai de réponse des opérateurs et à la facilité de réalisation d'analyses de sensibilité. Les choix des situations initiales, des événements à introduire lors du transitoire pour parcourir tous les pas de la consigne et le dépouillement des résultats sont pour le moment manuels. Dans ce développement, ce sont les procédures de conduite, les automatismes, les instruments et le procédé qui sont simulés.

Dans l'énergie et en aéronautique, des simulateurs « pleine échelle » sont aussi utilisés pour la validation des postes de conduite et des salles de commande puis pour la formation à la conduite normale et sur incidents. Toutes les parties du système de contrôle-commande de la figure 4 sont alors simulés [2].

Dans l'industrie électrique et les transports, une évolution récente consiste à utiliser ces possibilités de simulation pour une réception virtuelle des installations. Ceci conduit à représenter de façon très fine les systèmes de contrôle-commande, non pas en les simulant, mais en émulant les logiciels réels des applications et des fonctions de base des équipements utilisés. Dans l'énergie, la mécanique de la centrale est simulée, le contrôle-commande (automatismes et salle de commande) n'est pas simulé mais implanté et utilisé sur un calculateur, c'est-à-dire que ce sont les vrais logiciels de base et applicatifs (et leurs paramètres) qui sont exécutés. Il est alors possible de réceptionner en usine le développement complet du contrôle-commande principal et de la salle de commande sur une centrale virtuelle [1].

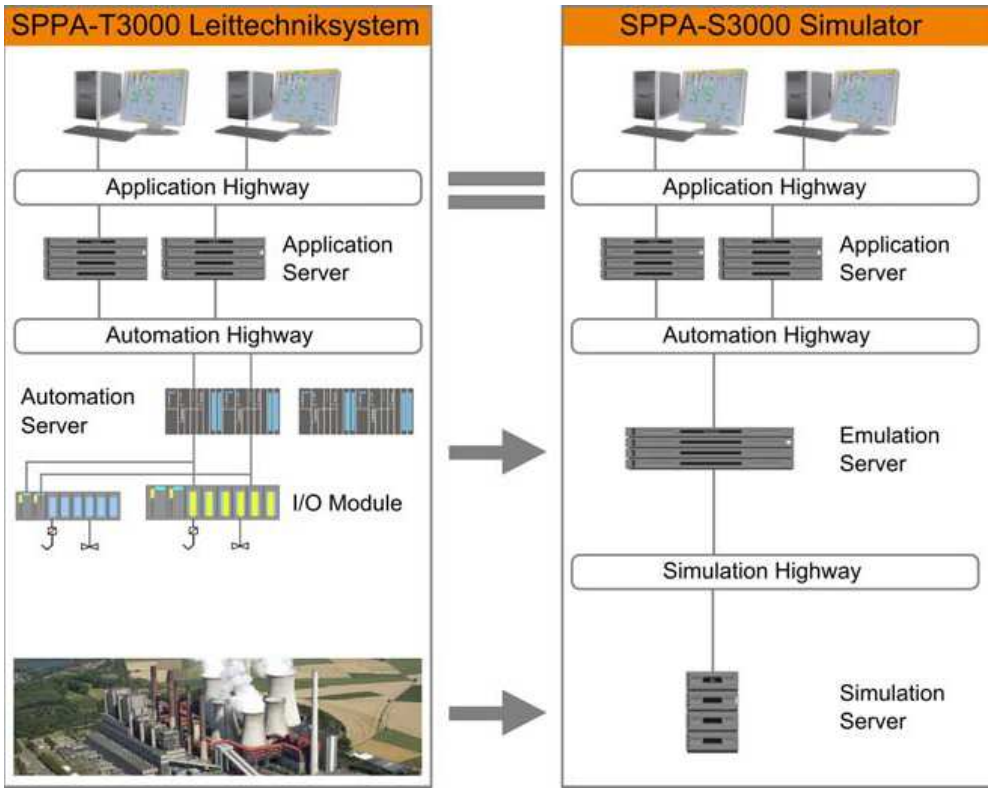


Figure 5 : Configuration de la simulation pour une mise en service virtuelle dans l'énergie [1]

A gauche l'architecture du système réel. A droite, le système émulé : toute la partie niveau 2 (au-dessus du réseau automate « Automation Highway » est constituée du système réel de conduite et s'interface avec la partie émulée (niveau 1 : les automates) et simulée (niveau0 et procédé).

2.2.2. Niveau de détail des modèles

Ce qui précède indique que, suivant les objectifs du test et les systèmes à tester, on peut utiliser des modèles de différents niveaux de représentativité dans les simulations. Dans une représentation de plus en plus fine des systèmes, on distingue, dans le test de validation par simulation, plusieurs niveaux de détail, qui pourront être utilisés de moins en moins tôt dans le développement :

- Pour les automatismes du contrôle-commande :
  - Les modèles et spécifications fonctionnelles exécutables (blocs fonctionnels), on parle en Ingénierie des Systèmes de MIL (« Model in the loop »)
  - Le code des applications des automatismes, SIL (« Software in the loop »)
  - Le code des applications des automatismes et une simulation des logiciels de base des équipements et de l'architecture (fonctionnement simultané de plusieurs unités de traitement avec les cartes d'entrées/sorties et les mécanismes d'échanges d'informations, temps d'exécution des différentes tâches de l'application et temps de réponse des différentes actions d'entrées / sorties, ordonnancement des tâches et instructions des unités centrales, etc.)
  - Les logiciels réels des applications par une émulation des applications des automatismes et des logiciels de base plus une simulation réduite des fonctions de base des équipements (essentiellement une représentation des parties matérielles des entrées / sorties et des supports physiques de communication)
  - Le système de contrôle-commande réel programmé avec les applications, HIL (« Hardware in the loop »).

Note : pour les configurations « In-the-Loop » (MIL, SIL, HIL) la notion de boucle fait référence au fait que le contrôle-commande est connecté en boucle fermée avec un modèle de procédé/parties opératives.

Le degré de finesse de la modélisation peut aussi varier dans les différentes catégories listées ci-dessus. On pourra par exemple modéliser ou non les schémas-types dans la partie contrôle-commande.

- Pour les instruments de la partie opérative et pour la partie mécanique :
  - Des temporisations
  - Des fonctions de transfert (par exemple des transformées de Laplace)
  - Des blocs de simulation du comportement physique et des équations

Les simulateurs qui modélisent des équations physiques peuvent aussi représenter plus ou moins finement la réalité, par exemple des écoulements en une ou trois dimensions pour les systèmes fluides, monophasique ou diphasique ou air / eau. L'étude de certains phénomènes nécessite des simulateurs multi-échelles et multi-physiques (électricité, thermodynamique, mécanique, neutronique…).

2.3. Verrous scientifiques et techniques de la validation par simulation de partie opérative des systèmes de contrôle-commande critiques

Les principales difficultés scientifiques et techniques pour valider par simulation des systèmes de contrôle-commande critiques sont liées au grand nombre de cas d'étude à produire, pour un système

de contrôle-commande qui peut aussi être complexe[4]. Ceci conduit à envisager une automatisation dans l'utilisation des simulateurs et dans le dépouillement des résultats, qui oblige à lever les principaux verrous scientifiques suivants :

- Clarifier le problème et l'objectif d'une automatisation des simulateurs pour la validation des systèmes de contrôle-commande critiques ;
- Choisir une méthodologie d'utilisation des simulateurs, parmi celles disponibles, qui permette d'atteindre cet objectif ;
- Proposer une stratégie de production automatique de cas d'étude qui évite l'explosion combinatoire du nombre de cas potentiellement simulables ;
- Proposer une stratégie d'analyse automatisée des résultats de simulation de ces cas d'étude.

Ces verrous devraient être normalement en grande partie levés par ce qui précède et ce qui suit. S'agissant d'un secteur d'activité critique, la simplicité des solutions proposées a été recherchée.

De plus, certains formalismes, modèles et méthodes du développement proposé font l'objet d'un approfondissement théorique et méthodologique par les partenaires académiques du projet Vacsim dans les autres tâches du programme.

2.4. Stratégies proposées dans l'utilisation des simulateurs pour les scénarios de test

La méthode d'utilisation des simulateurs à investiguer dans le projet vise à proposer une automatisation à la fois :

- De la définition des scénarios de test, dont l'initialisation des simulateurs, le choix des incidents et l'introduction de défauts lors du transitoire simulé ;
- De l'analyse des résultats, dont la vérification des performances, du respect des propriétés de sûreté de fonctionnement et de l'atteinte des objectifs des trajectoires.

Ces deux thématiques pourront mettre à profit les travaux des partenaires académiques du projet sur la validation formelle de propriété quantitative, pour proposer des algorithmes augmentant le niveau d'automatisation dans l'utilisation d'un simulateur et l'analyse de ses résultats. Il s'agit de spécifier un « superviseur de la simulation » dans une première définition décrite ci-après. Cette spécification fera l'objet d'un premier développement du livrable L2.2 du projet. Ce développement sera ensuite complété par l'introduction d'algorithmes supplémentaires, augmentant l'automatisation de l'utilisation des simulateurs.

2.4.1. Taux de couverture des scénarios de test par simulation

Le test par simulation de la commande des systèmes critiques conduit à utiliser plusieurs critères de taux de couverture :

- Des états de la partie contrôlée ;
- Des incidents sur la partie contrôlée et sur le contrôle-commande ;
- Des fonctions d'automatismes et d'IHM utilisées ;
- Des valeurs des variables et des traitements logiques et analogiques des automatismes ;
- Des procédures et des branches des procédures de conduite normale et sur incident.

Il est proposé d'implémenter, dans un premier temps, des algorithmes de calcul de ces taux de couverture dans le développement du livrable 2.2 du projet, afin de rendre compte de la proportion

[4] Du fait de redondances et d'isolements entre des parties de criticités différentes, mais parfois aussi d'un nombre conséquent d'équipements et de composants concernés.

des différentes parties du système testées par simulation.

Dans un second temps, la recherche d'un scénario de simulation permettant d'atteindre un objectif de test exprimé par un taux de couverture pourra donner lieu à l'ajout d'algorithmes, plus ou moins complexes suivant le critère concerné, qui seront à développer dans le projet. Des premières propositions de ces techniques à discuter dans le projet sont abordées dans le chapitre 2.3.2 suivant.

2.4.1.1. Validation fonctionnelle

Dans une activité de validation fonctionnelle au plus tôt des spécifications des fonctions du contrôle-commande ou de première utilisation d'un système numérique de contrôle-commande une fois programmé, la partie mécanique peut être simulée de façon très simplifiée par bouclage des sorties du système à ses entrées, en ajoutant éventuellement des retards dans ces boucles. Le critère de taux de couverture généralement utilisé est la proportion des blocs de traitement et des fonctions d'interfaces du système testés. Cette activité n'est plus évoquée dans la suite de ce livrable du projet, du fait de la très grande simplicité de la représentation de la partie contrôlée.

2.4.1.2. Etats de fonctionnement normaux

Un premier critère de taux de couverture à surveiller dans l'utilisation des simulateurs est la proportion des états de fonctionnement normaux et stables du produit ou de l'installation testés (utilisés dans les scénarios de test par simulation).

2.4.1.3. Procédures de conduite

La proportion du nombre de procédures de conduite normale et sur incident exécutées manuellement ou automatiquement en simulation est un critère classique. Un second critère concerne la proportion de branches de ces consignes parcourues dans les scénarios de test.

Le nombre d'objets d'IHM utilisés dans l'utilisation manuelle de ces procédures de conduite est un critère supplémentaire, mais les exigences réglementaires ou normatives imposent en plus de pouvoir tracer les scénarios de test et les parties des procédures ayant conduit à utiliser chaque objet d'IHM [6].

2.4.1.4. Fonctions d'automatisme

Fonctions logiques de protection et d'alarmes

Le critère le moins contraignant concerne l'activation au moins une fois de chaque fonction de protection et l'élaboration au moins une fois de chaque alarme.

Un critère classiquement utilisé concerne le passage de 0 à 1 et de 1 à 0 de chaque signal logique en sortie de chaque bloc de spécification.

Un critère plus exigeant peut être l'atteinte de la combinatoire logique en entrée de chaque bloc logique (00,01,10,11 en entrée d'un « OU » logique par exemple).

Pour le test des systèmes temporisés, on contrôle a minima l'activation et l'atteinte de l'échéance des temporisations.

Régulations

L'utilisation de chaque régulation en modes manuel et automatique dans les tests doit être surveillée.

La valeur d'une variable analogique (par exemple la mesure d'une grandeur physique régulée) peut varier à l'intérieur d'un intervalle [min, max] correspondant à des extremums électriques ou de stockage en mémoire pour une variable numérique.

Une variable analogique est généralement utilisée dans un intervalle [min_g, max_g] plus petit, qui correspond à des valeurs significatives de la variable dans son utilisation pour des fonctions de contrôle-commande. Quand la valeur se situe dans les intervalles [min, min_g] ou [max_g, max] on dit que la variable est hors gamme. Les spécifications précisent les gammes dans lesquelles les grandeurs surveillées doivent rester. Le critère selon lequel une variable analogique reste dans un domaine donné correspond à une propriété qu'il faudra vérifier lors du déroulement des scénarios de

test.

Les applications de contrôle-commande conduisent souvent à découper l'intervalle de validité de la mesure en domaines délimités par des seuils (d'alarmes, d'autorisation de commutation ou d'automatisme de basculement en mode automatique ou manuel, etc.). Pour un cas courant d'une variable instrumentée surveillée par deux seuils bas et deux seuils hauts d'alarmes, les domaines de variation des valeurs peuvent être représentés par des intervalles [min, min_g, $seuil_{tbas}$, $seuil_{bas}$, $seuil_{haut}$, $seuil_{thaut}$, max_g, max]. Quand le produit ou le processus est dans un état qui conduit à l'utilisation de la variable, sa valeur se situe en fonctionnement normal dans la plage [$seuil_{bas}$, $seuil_{haut}$].

Un premier critère de taux de couverture dans l'utilisation d'une variable analogique concerne l'appartenance de sa valeur aux différents domaines délimités par des seuils [S1, S2], y compris les intervalles [min, min_g] ou [max_g, max] pour des tests de robustesse des applications à l'invalidité de variables hors gamme. Un critère à utiliser pourrait être l'appartenance de la valeur au voisinage de la moyenne des seuils, du type (S1-S2) / 2 +/- x% (S1 – S2), x étant paramétrable.

Un second critère concerne l'utilisation de valeurs au voisinage des seuils (Si +/- y % Si, y étant paramétrable) afin de vérifier le calage des seuils.

Un troisième critère concerne l'évolution continue d'un domaine à un autre, en franchissant un seuil.

Une évolution continue de la valeur entre les deux bornes min et max est appelée pleine gamme ou pleine échelle, la surveillance de l'utilisation d'une telle évolution dans un des scénarios de test par simulation peut correspondre à un critère supplémentaire.

Remarque : La réalisation d'un test où la valeur d'une variable du contrôle est min ou max peut être plus difficile à réaliser dans une co-simulation qu'en test en boucle ouverte car elle peut correspondre à des états physiquement inaccessibles donc plus difficilement simulables.

Séquences automatiques de démarrage et d'arrêt

Comme pour les fonctions logiques, le premier critère est l'utilisation de la séquence au moins une fois.

Un second critère concerne l'essai d'utilisation de la séquence en l'absence de chacune des conditions initiales d'utilisation de la séquence ou en cas d'indisponibilité des fonctions ou matériels requis pour son utilisation (si la séquence automatique est bien faite, l'ordre d'utilisation devrait dans ces cas conduire à un refus).

2.4.1.5. Incidents mécaniques et défaillances du contrôle-commande

Trois critères au moins d'utilisation de défauts correspondant à la couverture des incidents de la partie mécanique et des défaillances du contrôle-commande prévus à la conception sont à surveiller pour :

- Les incidents sévères de la mécanique ayant été considérés lors de son dimensionnement ;
- L'ensemble des incidents mécaniques pris en compte à la conception (ex. : dérive d'un capteur) ;
- Les défaillances prévues du contrôle-commande (ex. : perte d'un rack d'entrées/sorties).

2.4.2. Définition des scénarios de test par simulation

2.4.2.1. Choix d'un état initial

Un produit complexe ou une installation industrielle peuvent comprendre plusieurs états normaux et stables de fonctionnement. Il s'agit de pouvoir choisir un de ces états comme situation de départ d'un scénario de simulation.

2.4.2.2. Exécution des procédures de conduite normale et simulation des transitoires normaux d'exploitation

Il s'agit de valider le bon fonctionnement normal par :

- La simulation de l'utilisation des procédures de conduite normale (démarrage et arrêt, changement d'états initiaux définis au chapitre précédent) une fois transformées en une version exécutable ;
- La simulation des grands transitoires normaux d'exploitation (îlotage, changement de régime de puissance…).

L'objectif est la validation fonctionnelle de la conduite prévue dans ces procédures. Il faut également vérifier le non-déclenchement intempestif des protections du système critique de contrôle et l'absence d'apparition d'alarmes au cours du déroulement des scénarios de test.

2.4.2.3. Etude de sensibilité

Dans un état initial, chaque paramètre a une valeur fixée. Pour une liste de paramètres sélectionnés, il s'agit de pouvoir rejouer n scénarios de test par simulation à partir de n états initiaux déduits de cet état initial. Un état est déduit par une variation d'un des paramètres de la liste en transformant sa valeur V par une valeur V_i égale à V + k_i x% V, ki appartenant à [-1 ; +1] et i à (1, 2,..,n) en découpant l'intervalle [-1 ; +1] en (n-1) domaines de valeurs égaux. Les valeurs physiquement non représentatives doivent être éliminées du traitement (pressions ou niveaux de bâche négatifs par exemple). Les valeurs x et n doivent pouvoir être choisies pour un paramètre ou un groupe de paramètres.

Note : cette fonctionnalité a l'avantage de la simplicité mais rend nécessaire un travail d'analyse dans la modélisation pour définir les paramètres pouvant varier individuellement dans une étude de sensibilité. Par exemple, pour un réacteur de centrale nucléaire, une légère variation de la pression ou des températures peut correspondre à une situation de fonctionnement réaliste. A l'inverse, une variation de pression d'un pressuriseur ne peut s'envisager de façon réaliste qu'en faisant simultanément varier la température, les deux variables étant liées par la courbe de saturation de l'eau en milieu diphasique. Le projet VACSIM devrait être l'occasion pour les partenaires de réfléchir à une méthodologie de choix des paramètres pouvant faire l'objet de ce traitement.

2.4.2.4. Choix d'un défaut mécanique

Il s'agit d'un des problèmes principaux dans la définition des scénarios de test. Le chapitre propose des fonctions de difficulté croissante, afin de réaliser un démonstrateur par étapes dans le lot 2.2 du projet.

Incidents de dimensionnement

Un produit complexe ou une installation industrielle peuent avoir été conçus en prenant en compte des incidents sévères dans leur dimensionnement. Ces cas de fonctionnement, en nombre limité, sont souvent difficilement reproductibles en situation réelle sur le vrai système sans l'endommager. Cela renforce l'intérêt de la simulation de ces incidents pour valider les systèmes de contrôle critiques prévus pour les maîtriser. Il s'agit de pouvoir choisir un de ces incidents de dimensionnement, normalement associé à un état initial donné, pour l'introduire dans le modèle de partie mécanique et simuler le transitoire faisant intervenir le contrôle critique à tester.

Le démonstrateur du livrable 2.2 devrait permettre de rejouer facilement l'ensemble des incidents de fonctionnement, en automatisant l'étude de sensibilité prévue au paragraphe précédent et la surveillance de propriétés de sûreté de fonctionnement décrite au chapitre suivant. Cet environnement serait déjà d'un grand intérêt, notamment en cas de modification du système de contrôle ou en cas d'une demande d'un nombre de tests importants à produire pour une justification de la qualité d'un système de contrôle.

Incidents pris en compte à la conception et analyse des seuils d'alarmes et de protection

Les états initiaux du paragraphe précédent sont des états normaux de fonctionnement ou d'arrêt. Les

valeurs des variables du procédé sont donc dans des plages nominales qui ne correspondent pas à l'atteinte de seuils d'alarmes ou de protection.

Une extension de la démarche du paragraphe précédent est à investiguer, avec l'introduction une à une des situations d'incidents prévus à la conception dans un scénario simulant le transitoire. La définition manuelle de ces situations à simuler est envisageable, pour une installation ayant bénéficiée lors de sa conception d'une chaîne de CAO et d'un système d'information modernes et performants permettant de codifier toutes ces situations d'incidents prévues sur la partie mécanique. Pour un produit complexe et coûteux ou une installation industrielle critique, ce devrait être la démarche normale. Si ces situations ne sont pas codifiées ou accessibles, par exemple si elles sont uniquement décrites par des textes, la définition à la main de plusieurs milliers d'incidents à introduire dans le scénario de simulation risque de devenir rapidement fastidieuse.

Le projet devrait permettre d'investiguer plusieurs approches pour une automatisation, si nécessaire, de cette définition d'incidents à utiliser pour les tests de validation par simulation.

Le plus simple, mais aussi le moins réaliste, consiste à faire évoluer systématiquement, dans un transitoire, chaque variable du procédé concernée par une alarme ou une protection, à partir de l'état initial, jusqu'à atteindre puis dépasser les seuils d'alarmes et de protection.

La vraisemblance du phénomène physique simulé n'est pas garantie car ces variables sont le plus souvent endogènes[5]. La possibilité d'une recherche automatique d'une évolution des variables de commande du procédé ou d'une introduction d'un défaut conduisant à cette évolution de la variable instrumentée considérée est à discuter dans le projet. Il s'agirait, par exemple, de retrouver des conditions du type arrêt d'une pompe, fermeture d'une vanne réglante ou introduction d'une fuite, conduisant à simuler une baisse de débit jusqu'à atteindre des seuils d'alarmes et de protection sur bas débit. La propagation symbolique de l'évolution souhaitée de la variable instrumentée via les équations physiques du modèle est à envisager. Dans le projet VACSIM, l'I3S étudie aussi les apports potentiels des techniques de BMC ("Bounded Model Checking") basées sur la programmation par contraintes. Dans cette approche, la partie contrôle et/ou les équations physiques du modèle sont décrites par des programmes impératifs comportant des calculs en virgule flottante. Ces programmes sont eux-mêmes transformés en systèmes de contraintes non linéaires sur les nombres à virgule flottante. La résolution de ce type de systèmes de contraintes nécessite des solveurs spécifiques qui garantissent la correction des solutions sur les nombres à virgule flottante.

Une démarche systématique donc automatisable peut consister à définir non pas les incidents prévus pour chaque équipement mais, de façon générique, de définir des incidents types par classe d'équipements standards (fuite de bâche, arrêt de pompes, ouverture ou fermeture intempestive de vannes, blocage de vannes réglantes). Il s'agirait ensuite d'instancier ces types d'incident en cas d'incident particulier de chaque matériel concerné pour proposer un cas à tester en simulation.

Situations et incidents permettant l'atteinte d'un critère de couverture des blocs de contrôle ou le non-respect d'une propriété

Les apports de la programmation par contraintes pour la vérification de propriétés ont été étudiés dans le projet ANR TESTEC. Ces travaux ont permis d'élaborer des techniques de génération de contre-exemples qui s'appuient sur différentes heuristiques de parcours du graphe de flot de contrôle des systèmes vérifiés [14]. Cette technique a été mise en œuvre avec succès sur une étude de cas fournie par Geensys / Dassault Systèmes [15, 16]. Une extension à la proposition d'initialisation de variables de commande du simulateur est à discuter dans le programme VACSIM.

D'autres travaux, en particulier au CEA, ont étudié l'utilisation de la programmation par contraintes pour la génération de tests permettant d'atteindre des objectifs de test décrits en Lustre [3] [4]. Ces techniques ne sont pas triviales. Leur adaptation à l'utilisation de simulateurs pourrait être discutée dans le projet VACSIM.

L'atteinte d'un taux de couverture des blocs d'une spécification du système de contrôle n'est pas en

[5] Les variables exogènes sont déterminées hors du système d'équations de la partie mécanique modélisée, alors que les variables endogènes sont déterminées par le système d'équations.

soit un objectif de bon fonctionnement de la partie contrôlée. La non-sollicitation d'une partie du contrôle dans la simulation de tous les cas de fonctionnement normaux et toutes les situations d'incidents prévus à la conception pourrait amener à se poser la question de la qualité ou de l'intérêt de cette partie. C'est donc bien la fonction de surveillance de ces taux de couverture qui est à traiter de façon prioritaire dans le projet VACSIM.

A réfléchir : génération de transitoire pour l'inversion de tendance des commandes analogiques, basculement auto/manu des régulations

Défaillances prévues du contrôle-commande

Comme indiqué dans ce qui précède, il est proposé d'éviter la combinatoire résultante de la multiplication des cas possibles de défaillance de la partie mécanique et de défaillance de la partie contrôle. Il s'agirait de rejouer des scénarios de test d'un incident mécanique en injectant des défauts sur les seules parties du contrôle-commande sollicitées lors du transitoire résultant de l'incident.

Les techniques à mettre en œuvre pour identifier les parties sollicitées ainsi que le choix d'un défaut sur ces parties de contrôle sollicitées lors d'un transitoire (instruments, automatismes et IHM) sont à discuter dans le projet.

Scénarios d'utilisation des procédures de conduite sur incident

Il s'agit d'initialiser le simulateur puis de simuler l'incident conduisant à l'utilisation d'une procédure transformée en un module exécutable. Une recherche automatique des défaillances à introduire dans le scénario du transitoire pour couvrir toutes les branches de la procédure, à partir d'une analyse des pas de test de la procédure, est à développer dans le projet.

2.4.3. Définition de la sanction du test par simulation

Ce chapitre concerne la définition de l'oracle du test.

Surveillance de propriétés de sûreté de fonctionnement

EDF R&D, Dassault Aviation, Dassault Systèmes ont étudié la modélisation de propriétés de sûreté de fonctionnement à l'occasion du projet européen ITEA 2 'EuroSysLib' [7]. Le démonstrateur L2.2 du projet devrait déjà intégrer ces résultats ainsi que les possibilités de contrôle et la bibliothèque des propriétés développées sous Modelica. Les propriétés surveillées peuvent concerner aussi bien la partie mécanique (seuil de pression à l'aspiration d'une pompe en fonctionnement pour éviter une cavitation par exemple) que le contrôle-commande (absence de surcharge des CPU ou des réseaux de communication par exemple).

L'objectif était d'éditer et de surveiller des propriétés. Chaque propriété est caractérisée par la réponse aux trois questions :

- **Où** ? : dans quelle partie du système la propriété doit être évaluée. Par exemple, pour une propriété de sûreté sur une pression, cela se situera au niveau du capteur de pression correspond.
- **Quand** ? : à quel moment la propriété doit être évaluée. Cela peut correspondre notamment à des états différents du système. Par exemple la propriété doit être vérifiée quand le système est en fonctionnement à pleine puissance uniquement, ou pour une pompe, la propriété de non-cavitation doit être vérifiée tout le temps.
- **Comment** ? : il s'agit de déterminer quels sont les critères qui vont permettre d'évaluer la propriété. Cela intègre des données physiques, mais également des données sur l'état du système. Cela peut également intégrer des traitements supplémentaires non implémentées dans le modèle ou dans le contrôle-commande. Par exemple, s'il faut vérifier qu'une grandeur physique ne varie pas trop vite, il faudra rajouter un calcul de gradient afin de pouvoir évaluer la variation de la grandeur.

Les travaux ont notamment permis d'éditer des propriétés d'appartenance à des domaines de fonctionnement ou de respect de gradients d'évolution de paramètres. Toutefois, ils ont aussi mis en évidence des limites dans la manipulation d'opérateurs temporels. Il est par exemple difficile

d'exprimer de façon déclarative l'appartenance de paramètres à un domaine de fonctionnement à une échéance donnée.

Les outils basés sur Modelica ne permettent pas en effet de décrire simplement un tel critère ou même le calcul de temps cumulé de fonctionnement d'un actionneur. Cela reste cependant possible à partir des éléments de bibliothèques élémentaires (compteurs temporels, opérateurs logiques de base, etc.) mais peut aboutir à une programmation complexe.

Un modèle Statechart a aussi été développé pour sélectionner les modèles de propriétés et de fonctionnement suivant les états simulés et les conditions d'environnement (donc suivant le scénario de test simulé). En effet, pour un système donné :

- Le nombre et le type de propriétés à satisfaire peuvent différer selon le mode de fonctionnement étudié (par exemple, en fonctionnement normal, une exigence peut être d'évacuer une certaine quantité de chaleur alors que celle-ci est relâchée en fonctionnement incidentel ou lors de conditions météorologiques extrêmes) ;
- Bien souvent, non pas un mais plusieurs modèles sont utilisés pour représenter les phénomènes physiques de modes de fonctionnement différents (par exemple, les outils de simulation pour le fonctionnement normal sont en général différents de ceux utilisés dans le cadre d'études incidentelles / accidentelles).

Ce point est détaillé dans le paragraphe §3.

Atteinte des objectifs de conduite et validation de la conduite normale

Dans l'exécution des procédures de conduite normale et des transitoires normaux d'exploitation il s'agit a minima de vérifier le non-déclenchement intempestif des protections du système critique de contrôle, l'absence d'atteinte de seuils d'alarmes et d'incidents sur le procédé (sollicitation de soupapes mécaniques de protection contre les surpressions, perte de fluide, débordement de bâche…).

Respect des trajectoires et séquences temporelles

Le démonstrateur L2.2 du projet devrait aussi permettre la vérification de l'atteinte des objectifs des consignes et séquences de conduite. Certains objectifs sont exprimables sous forme d'appartenance de paramètre de fonctionnement à un état final objectif de la consigne ou séquence. Le temps d'atteinte de cet état final doit a minima être surveillé.

Pour d'autres objectifs de conduite ou pour la surveillance du comportement normal d'une installation ou d'un produit lors de transitoires répertoriés, il peut être utile de définir un programme prévu d'évolution de paramètres (du type : dans telle fenêtre de temps, telle valeur et sa dérivée doivent appartenir à telles plages de valeurs). L'automatisation de l'oracle pour ce point devrait pouvoir se réduire à la surveillance dans le transitoire de l'exécution d'une séquence temporelle d'événements représentable par des automates temporisés [11]. L'idée est d'utiliser des automates temporisés pour décrire des propriétés temporelles, et ensuite de vérifier sur les traces d'exécution (online ou offline) que ces traces ne violent pas les propriétés exprimées.

Pour cela, on peut envisager de s'inspirer des travaux sur le test passif ou le monitoring de systèmes temporisés, comme [12] et [13]. L'outil AMT [12] est un outil permettant de vérifier des propriétés temporelles sur des signaux continus avec comme logique d'entrée STL/PSL. Dans le même esprit, LARVA [13] est un outil de monitoring qui prend en entrée des propriétés de sûreté exprimées dans des notations variées (par exemple Lustre) et qui traduit ces propriétés en une variante des automates temporisés.

La tâche 4 du projet VACSIM prévoit de travailler sur ces aspects monitoring de systèmes temporisés et de les étendre avec des techniques d'enforcement.

Taux de couverture

Le démonstrateur L2.2 du projet devrait permettre de surveiller l'ensemble des critères du chapitre 2.3.1.

Autres vérifications

Le démonstrateur L2.2 du projet n'est pas l'occasion d'un développement important pour l'étude d'une propriété complexe spécifique à un métier comme le calcul de contraintes mécaniques et sa visualisation. La connexion du démonstrateur à des environnements existants de ce type devrait toutefois être discutée dans le projet.

L'Université de Stuttgart a récemment défini des indicateurs de performance des régulations [8]. La surveillance de ces indicateurs pourrait être intéressante dans une phase de validation de schémas de régulation par simulation.

3. Spécification et architecture d'un superviseur de la simulation

L'objectif de ce paragraphe est de spécifier une première version d'un moteur d'exécution des simulations et d'analyse de leurs résultats.

Les fonctionnalités de base de cet environnement concernent l'initialisation des états, le choix d'un scénario, l'analyse de sensibilité, la vérification des propriétés et de l'atteinte d'objectifs, la surveillance de taux de couverture, le rapport automatique des résultats de tests (quelles sont les propriétés non-vérifiées ? à quel moment et pendant combien de temps ?…), mais aussi la gestion des configurations des modèles et des tests,

Sur la base des travaux du projet européen EuroSysLib, nous proposons une architecture de simulateur dans lequel on distingue différents sous-systèmes qui ont été présentés dans les paragraphes précédents. Dans le cadre de la production d'énergie et des études dans EuroSysLib et dans le projet R&D INCOME (Ingénierie des modifications du contrôle-commande par les modèles), le périmètre du simulateur concerne un système élémentaire ou une sous-partie. L'ensemble de la centrale électrique n'est donc pas modélisée. Ces sous-systèmes sont les suivants.

- Le modèle du procédé / parties mécaniques
- Le modèle du contrôle-commande (loi de commande et/ou système support)
- Le modèle de l'environnement du système étudié
- Le modèle de propriétés qui intègre les propriétés qui doivent être vérifiées au cours des différents scénarios de tests

Tous ces éléments sont orchestrés par un superviseur. Il s'agit d'un automate à états qui va lire en temps réel certaines valeurs du procédé et de l'environnement afin de déterminer quel est l'état du système. En fonction de l'état déterminé, il va sélectionner un modèle correspondant à cet état pour le modèle de procédé, du contrôle-commande et de propriétés. Le superviseur développé dans EuroSysLib permet de choisir ou de configurer les modèles de processus, de contrôle-commande, d'environnement et de propriétés en fonction de l'état de la tranche ou du système modélisé. Dans le projet VACSIM, il s'agira d'augmenter ces possibilités par l'automatisation des stratégies décrites au chapitre 2.3 précédent.

Le schéma ci-après montre comment les différentes sous-parties interagissent :

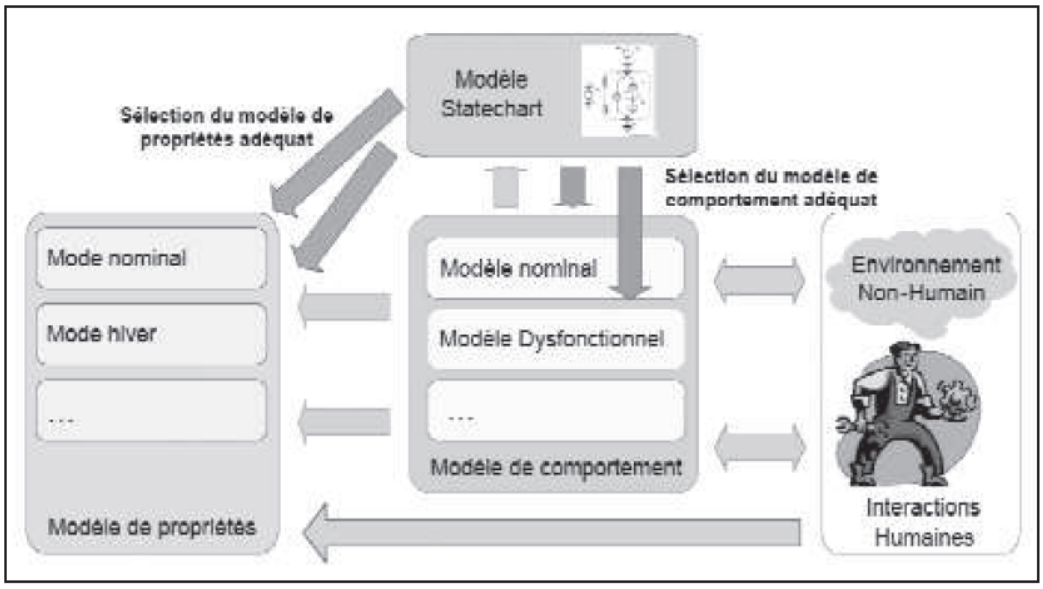


Figure 6 : architecture générale du simulateur orchestré par le superviseur

Au centre, le **modèle de comportement** est composé du modèle de procédé et du modèle de contrôle-commande. Dans une configuration HIL (Hardware-in-the-Loop) dans les phases plus avancées du projet, le contrôle-commande n'est plus simulé, il sort donc du modèle de comportement mais est en interaction avec le modèle de procédé.

On peut définir plusieurs modèles de comportements qui correspondent à différents états du système étudié :

- Un modèle nominal qui correspond au fonctionnement normal. Ce modèle peut lui-même être subdivisé en plusieurs modèles correspondant à des modes de fonctionnement différents qui impliquent un paramétrage différent voire une structure différente du modèle. Concernant le contrôle-commande, on peut prendre l'exemple d'un régulateur PID dont les paramètres sont optimisés différemment selon la puissance de la centrale électrique.
- Un ou des modèles de dysfonctionnement qui vont correspondre à l'introduction de défaillances lors des scénarios de tests.

Le **modèle de l'environnement** intègre toutes les interfaces avec les autres systèmes auxquels le système étudié est connecté. Cela peut se traduire par des conditions aux limites (par exemple valeur de consigne imposée) ou par une modélisation plus complexe de phénomènes physiques ayant un impact fort sur le système étudié. Le modèle d'environnement intègre aussi les interactions humaines (actions générées dans le cadre d'une procédure de conduite par exemple). Même si cela n'apparaît pas sur la Figure 6, le superviseur peut également lire des valeurs issues du modèle d'environnement. En effet, des variations significatives de l'environnement peuvent déterminer le basculement dans un état différent, c'est le cas par exemple si la consigne de puissance de la centrale est élaborée dans le modèle d'environnement.

Le **modèle de propriétés** comporte l'ensemble des propriétés qui ont pu être modélisées d'après les spécifications du système. Le modèle de propriétés peut être établi en deux étapes :

- Dans une première étape, les propriétés sont extraites de documents de spécifications où elles sont généralement décrites en langage naturel (texte). Elles sont alors présentées dans un tableau permettant de les répertorier et de commencer à les formaliser dans l'optique de pouvoir automatiser la vérification de celles-ci lors des tests MIL, SIL et HIL. Pour chaque propriété, on répond aux questions « Où ? » « Quand ? » « Comment ? » décrites dans le §2.3.3. L'étude de cas du paragraphe suivant donne un exemple d'un tel tableau.
- L'étape suivante consiste à programmer dans un langage qui permet de s'interfacer avec les modèles de comportement et d'environnement (dans l'exemple présenté au §4, il s'agit du

même langage pour ces trois modèles, à savoir Modelica). Le modèle de propriétés va lire en temps réel les valeurs issues du modèle de comportement afin d'évaluer les propriétés et émet un résultat booléen sur la validation ou non de la propriété. Le résultat est donc un profil temporel indiquant à quels instants la propriété a été vérifiée et à quels moments elle ne l'a pas été.

Le superviseur, en fonction de l'état qu'il a calculé va inhiber ou non la vérification de propriétés. Seules les propriétés pertinentes à vérifier dans l'état courant seront évaluées. Le superviseur peut comporter plusieurs états se situant à différents niveaux du modèle. Par exemple au niveau du système complet, il peut identifier un état global impactant l'intégralité des composants du modèle, dans le cadre de la production d'énergie cela peut être la puissance de la tranche. L'état peut aussi ne concerner qu'un composant, par exemple un actionneur peut être en marche ou à l'arrêt. Dans ce dernier cas, il apparaît que plusieurs propriétés n'auront pas besoin d'être évaluées.

4. Etudes de cas

4.1. Le système SRI « Système de Réfrigération Intermédiaire »

L'étude de cas choisi concerne le système élémentaire SRI des tranches nucléaires N4. Il permet d'amener l'eau de refroidissement brute des équipements du circuit secondaire à une température acceptable, au travers d'échangeurs thermiques et avec un circuit d'eau déminéralisée. Ce circuit présente certaines caractéristiques qui en font un cas test intéressant :

- Il comporte une régulation analogique avec la régulation de température qui se fait grâce aux vannes réglantes « Regulating valve 1 et 2 » et « By-pass valve » (cf. Figure 7 ci-dessous). Ces actionneurs en agissant sur le débit d'eau passant dans les échangeurs vont permettre d'évacuer plus ou moins de chaleur et donc de réguler la température.
- Une logique de contrôle à seuils avec la régulation de la bâche d'alimentation « Feeding tank » : ce traitement permet de faire l'appoint en eau dans le circuit SRI en compensant les éventuelles fuites. En fonction des dépassements des seuils min. et max. du niveau de la bâche d'alimentation, la vanne logique « Feeding on-off valve » s'ouvre ou se ferme.
- Une logique de démarrage de pompe. En fonctionnement normal deux pompes sont en service et assurent la circulation de l'eau avec un débit qui reste dans un domaine de fonctionnement donné. En cas de défaillance d'une des deux pompes, il y a démarrage automatique de la 3^{e} pompe. Les arrêts et démarrages des pompes peuvent également être effectués en mode manuel.
- Le système présente peu de dépendances avec d'autres systèmes, le modèle d'environnement est donc simple à modéliser (conditions aux limites)
- L'implémentation du modèle dans Dymola met en œuvre un nombre de variables limité ce qui limite la complexité du cas à traiter.

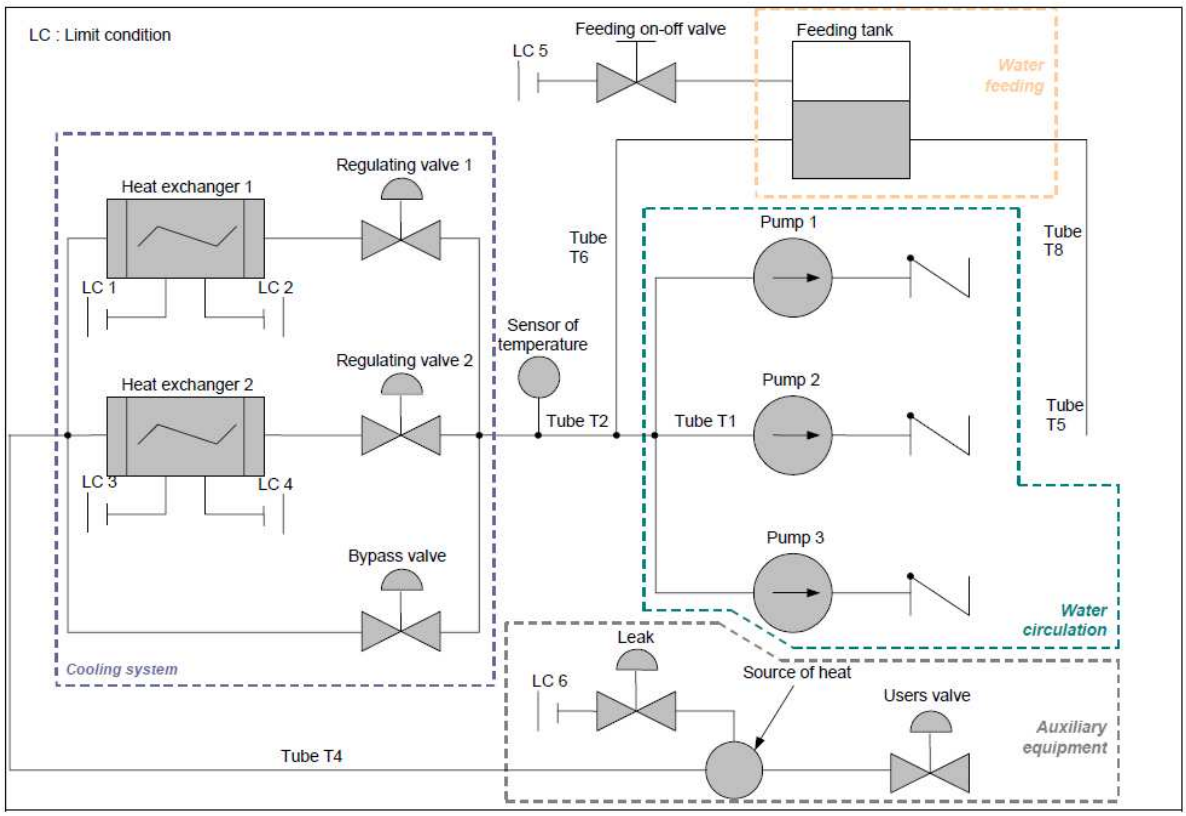


Figure 7 : schéma du fonctionnement du SRI (Circuit de réfrigération intermédiaire)

4.2. Le modèle de propriétés

A partir du DSE (Dossier de Système Elémentaire, documents de référence EDF spécifiant sous forme textuelle principalement le fonctionnement du système sous les angles fonctionnel, contrôle-commande, sûreté, matériel), la première version du modèle de propriétés a été élaborée sous forme d'un tableau. Les propriétés ont été typées par catégorie. Voici des exemples illustrant différents catégories de propriétés :

Au niveau sûreté de fonctionnement : « En mode normal de conduite, il ne doit pas y avoir moins de deux pompes en marche pendant plus de deux secondes (excepté en mode manuel). »

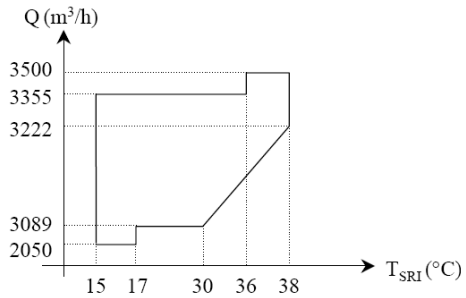
Au niveau matériel « En fonctionnement, une pompe ne doit pas caviter. »

Au niveau fonctionnel : « En mode normal de conduite, la température du circuit SRI doit se situer entre 15 ℃ et 38 ℃. »

Le tableau suivant présente l'intégralité des propriétés qui ont été listées :

N°	Description de la propriété	Opérateur utilisé	Données d'entrée	Status	Commentaire
1	Le transfert de chaleur doit être orienté vers le circuit SEN dans les échangeurs.	Seuil	**Variables:** - Différence de température entre SRI et SEN : Cooling.EchangeurAPlaques1D1.Ec.T - Cooling.EchangeurAPlaques1D1.Ef.T, idem pour l'échangeur 2 **Localisateurs**: aucun **Paramètres :** - différence de température minimale admissible (= 0)	Peut être implémenté	
2	La consigne de la température d'eau du SRI doit être maintenue à une valeur minimale de 17℃.	Seuil	**Variables:** - température de l'eau (consigne) : Control_Command.ReguTemperature1.Constante2 **Localisateurs** : aucun **Paramètres :** - consigne de température minimale admissible (Ts_min= 17℃)	Implémenté	
3	En mode de fonctionnement normal, la température d'eau du SRI doit être entre Ts - e et Ts + e (Ts : consigne de température).	Seuil	**Variables:** - température eau : sRI_system.cooling.CapteurT1.Tm - température eau (consigne) : Control_Command.ReguTemperature1.Constante2 **Localisateurs :** - fonctionnement normal : SRI_on et SEN_on et Temperate **Paramètres:** - écart admissible de température (e = +/- 1℃)	Implémenté	Cette propriété peut être utilisée pour évaluer la performance du contrôle-commande. La différence de température a été choisie de façon arbitraire, aucune valeur n'étant précisée.

N°	Description de la propriété	Opérateur utilisé	Données d'entrée	Status	Commentaire
4	En mode de fonctionnement normal, la température d'eau du SRI doit être entre 15℃ et 38℃.	Seuil	**Variables:** - température de l'eau : sRI_system.cooling.CapteurT1.Tm **Localisateurs :** - fonctionnement normal: SRI_on et SEN_on **Paramètres:** - température maximale admissible (= 15℃) - température maximale admissible (= 38℃)	Implémenté	
5	En mode de fonctionnement normal, le débit d'eau dans le circuit SRI doit être quasi constant (entre 3089 et 3355 m^3/h).	Seuil	**Variables:** - débit d'eau : sRI_system.water_Circulation.CapteurD1.Q **Localisateurs** : - fonctionnement normal: SRI_on et SEN_on et Temperate et PPunit_on **Paramètres**: - débits minimal et maximal admissibles (= 3089 et 3355 m^3/h)	Implémenté	La limite supérieure a été choisie arbitrairement de telle façon que la plage de variations du débit soit centrée sur le débit nominal (3222 m^3/h).

	Description de la propriété	Opérateur utilisé	Données d'entrée	Status	Commentaire
6	La température de l'eau et le débit doivent rester dans le domaine donné.	Inclusion dans un domaine	**Variables:** - débit d'eau : sRI_system.water_Circulation.CapteurD1.Q - température eau : sRI_system.cooling.CapteurT1.Tm **Localisateurs :** - mode de fonctionnement : SRI_on et not SEN_off **Paramètres:** - domaine (Q,T) 	Implémenté	Propriété rajoutée. Elle n'apparaît pas dans les spécifications mais permet d'implémenter un cas d'appartenance à un domaine à 2 dimensions.
7	Quand le SRI est en marche, la température de l'eau ne doit pas varier plus vite que 10℃/h.	Ramp	**Variables:** - température eau : sRI_system.cooling.CapteurT1.Tm **Localisateurs :** - Mode fonctionnement normal : SRI_on and SEN_on **Paramètres:** - débit minimal et maximal admissible (= 3089 et 3355 m^3/h)	Implémenté	Propriété rajoutée.

8	Quand le SRI est en marche, le débit d'eau ne doit pas varier plus vite que 10℃/h.10%/min.	Ramp	**Variables:** - débit d'eau : sRI_system.water_Circulation.CapteurD1.Q **Localisateurs :** - mode fonctionnement : SRI_on **Paramètres:** - Taux d'accumulation (= 10%)	Implémenté	Propriété rajoutée.

	Description de la propriété	**Opérateur utilisé**	**Données d'entrée**	**Status**	**Commentaire**
9	Quand le SRI est en marche, le débit d'eau doit être supérieur à 700 T/h.	Seuil	**Variables:** - débit d'eau : sRI_system.water_Circulation.CapteurD1.Q **Localisateurs :** - mode de fonctionnement : 2 pumps on **Paramètres:** - débit minima admissible (= 700 T/h)	Implémenté	De façon plus rigoureuse, cette propriété devrait être ajoutée au niveau du composant pompe. Elle est ici modélisée au niveau système dans la mesure où le débit minimal admissible n'est donné que pour le débit total au niveau du SRI.Scénario : débit nul
10	La pression ne doit pas excéder 8 bars.	Seuil	**Variables:** - pression de décharge de la pompe : sRI_system.water_Circulation.VolumeC1.Cs.P **Localisateurs:** aucun **Paramètres :** - pression maximale admissible (= 8 bar)	Implémenté	Cette propriété doit permettre d'éviter des sollicitations mécaniques excessive sur les composants du SRI. Scénario : débit nul

11	La pression dans le circuit SRI doit être égale ou supérieure à la pression dans le circuit SEN au niveau des échangeurs principaux si la pression du SEN est inférieure à 9 bars.	Seuil	**Variables:** - Pressions du SRI et du SEN dans les échangeurs : min(sRI_system.cooling.EchangeurAPlaques1D1.Pmc[i]) et max(sRI_system.cooling.EchangeurAPlaques1D1.Pmf[i]), i = 1 à 5, Idem pour l'échangeur 2 **Localisateurs**: - Quand SEN pressure (sRI_system.cooling.SourceP1.Pm) < 9 bar **Paramètres:** aucun.	Implémenté	Cette propriété vient de l'exigence suivante : les équipements doivent être refroidis par de l'eau déminéralisée, celle-ci ne doit pas être polluée par de l'eau brute en cas de fuite dans les échangeurs.
12	Le réservoir d'eau assure la circulation (par gravitation) pour bagues d'usure des TPA en cas de perte accidentelle de toutes les pompes.	Seuil	**Variables:** - débit d'eau : sRI_system.water_Circulation.CapteurD1.Q **Localisateurs:** - quand toutes les pompes sRI_system.water_Circulation.Failure_Pump = true (événement) **Paramètres:** - débit requis (= 20m^3/h) - durée	Non Implémenté	Il n'y a pas d'opérateur « tous » permettant de sélectionneur tous les actionneurs en une commande. Il est nécessaire de lister toutes les pompes. Le modèle n'est ici pas approprié pour le mode vidange.

	Description de la propriété	Opérateur utilisé	Données d'entrée	Status	Commentaire
13	En mode de fonctionnement normal, il ne doit pas y avoir moins de 2 pompes en marche pendant moins de 2s (à part pour les opérations manuelles).	Condition temporisée	**Variables:** 2 pompes sur 3 **Localisateurs :** - mode de fonctionnement normal : SRI_on et Temperate et PPunit_on **Paramètres:** - durée (= 2s)	Implémenté	Pas d'opérateur approprié dans Modelica.
14	La différence de températures dans l'échangeur doit être égale à 4.5 ℃ +/- 0.5℃ dans la configuration normale du SRI.	Seuil	**Variables :** - différence de température entre SEN et SRI : sRI_system.cooling.EchangeurAPlaques1D1.Tsc - sRI_system.cooling.EchangeurAPlaques1D1.Tef, idem pour l'échangeur 2 **Localisateurs:** - configuration normale : 2 pompes en marche et 2 échangeurs **Paramètres:** - plage de la différence de température : 4-5℃	Implémenté	Sans valeur donnée, la tolerance a été fixée arbitrairement
15	Le remplissage rapide (< 2 heures) du circuit SRI en cas de dysfonctionnement du système d'alimentation automatique.	Délai entre 2 évènements	**Variables :** - durée du remplissage manuel : temps entre l'ouverture de la vanne manuelle et l'atteinte du niveau ReguNiveau1.niveauMin1 par la variable FeedWater.Bache1.z **Localisateurs:** - défaillance du système d'alimentation automatique : si sRI_system.FeedWater.Failure_Valve_Feedwater = true **Paramètres** : - temps de remplissage maximal admissible (= 2 heures)	Non implémenté	Le critère de surveillance du remplissage d'eau a été choisi selon une vue physique même s'il n'est pas précisé dans les spécifications du SRI. Il n'y a pas d'opérateur approprié. Scénario "Remplissage rapide du SRI"

16	Le débit d'eau alimentaire doit être maintenu autour d'une valeur donnée.	Seuil	**Variables :** - débit d'eau : sRI_system.FeedWater.Bache1.Cs2.Q **Localisateurs:** - quand l'alimentation en eau est en mode automatique : sRI_system.FeedWater.VanneTOR1.Ouv = true Paramètres: - débit d'eau alimentaire admissible (= 30 m3/h) - tolérance (= +/- 2 m3/h)	Implémenté	Sans valeur donnée, la tolérance a été fixée arbitrairement

	Description de la propriété	**Opérateur utilisé**	**Données d'entrée**	**Status**	**Commentaire**
17	L'eau provenant du système SED doit être déminéralisée avec un pH égal à 7.	Seuil	**Variables :** - pH de l'eau venant du SED **Localisateurs:** aucun **Paramètres:** - pH admissible de l'eau d'alimentation (= 7) - tolérance	Non Implémenté	Le modèle n'est pas approprié (pas de modélisation du pH)
18	Le pH de l'eau du SRI doit être maintenu dans une plage de valeurs donnée.	Seuil	**Variables :** - valeur de pH de l'eau du SRI **Localisateurs:** aucun **Paramètres:** - pH minimal admissible(= 11) - pH maximal admissible (= 11.5)	Non Implémenté	Le modèle n'est pas approprié (pas de modélisation du pH)

L'implémentation de ce modèle de propriétés dans Dymola a ensuite été réalisée et connectée au modèle de comportement par des connecteurs de type multiplexeur de données.

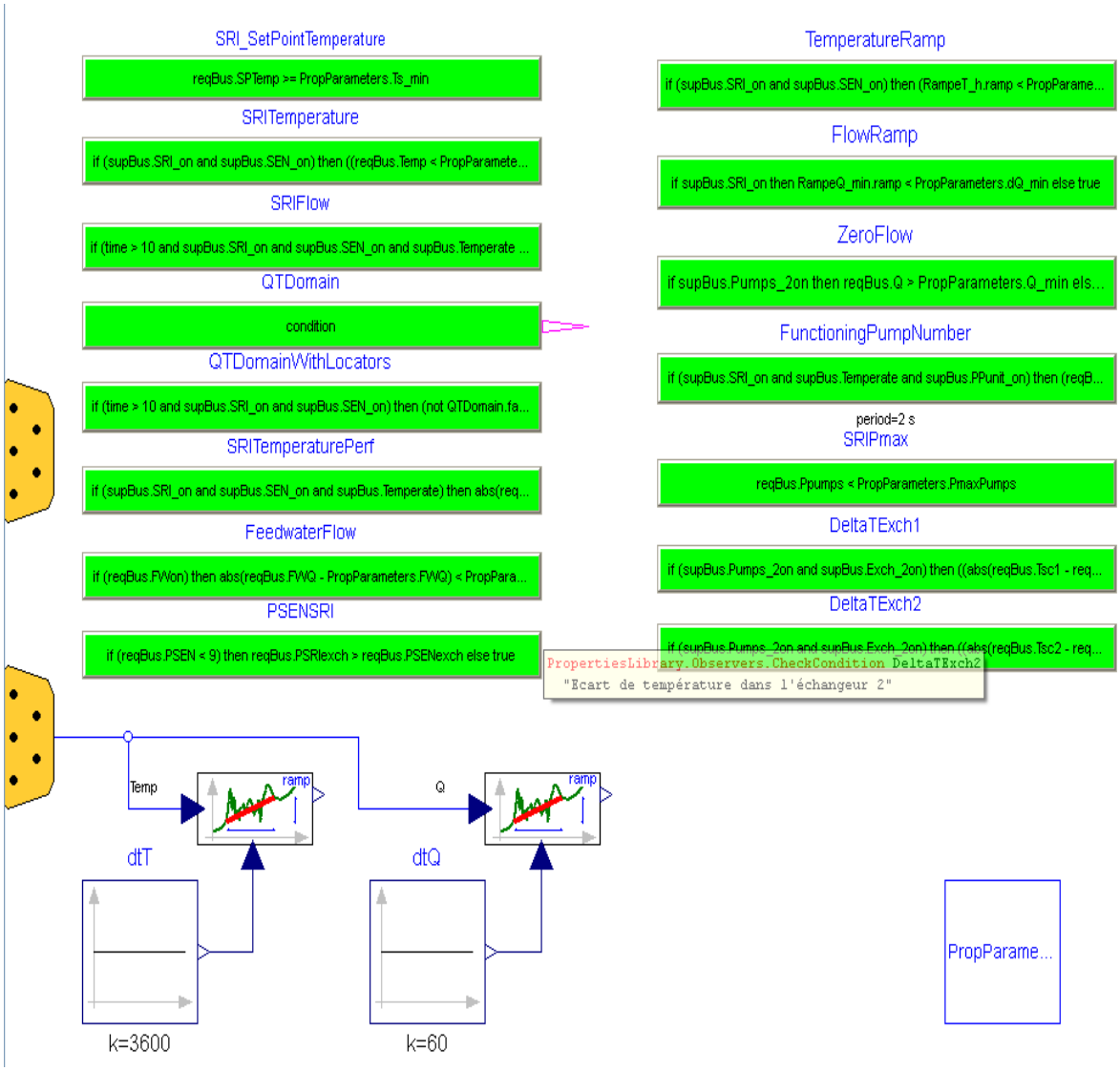


Figure 8 : modèle de propriétés implémenté sous Dymola

Chaque rectangle vert correspond à l'implémentation d'une propriété décrite dans le tableau ci-dessus. Les connecteurs jaunes sur le côté correspondent aux multiplexeurs qui alimentent le modèle de propriétés à partir des données de simulation du modèle de comportement. Les propriétés sont codées à partir des opérateurs logiques de base disponibles dans les bibliothèques Dymola.

Les blocs de traitements en bas de la figure illustrent des traitements complémentaires nécessaires à l'évaluation d'une propriété. Ici, l'évaluation du gradient de température et de débit a nécessité l'introduction d'un bloc dérivée.

4.3. Le modèle de comportement

Celui-ci a été établi également à partir de bibliothèques de base de Dymola ainsi qu'une bibliothèque développée par EDF qui intègre notamment des blocs Contrôle-Commande.

Le modèle de comportement est subdivisé en sous-fonctions liées au procédé et au contrôle-commande comme l'illustre la figure ci-dessous :

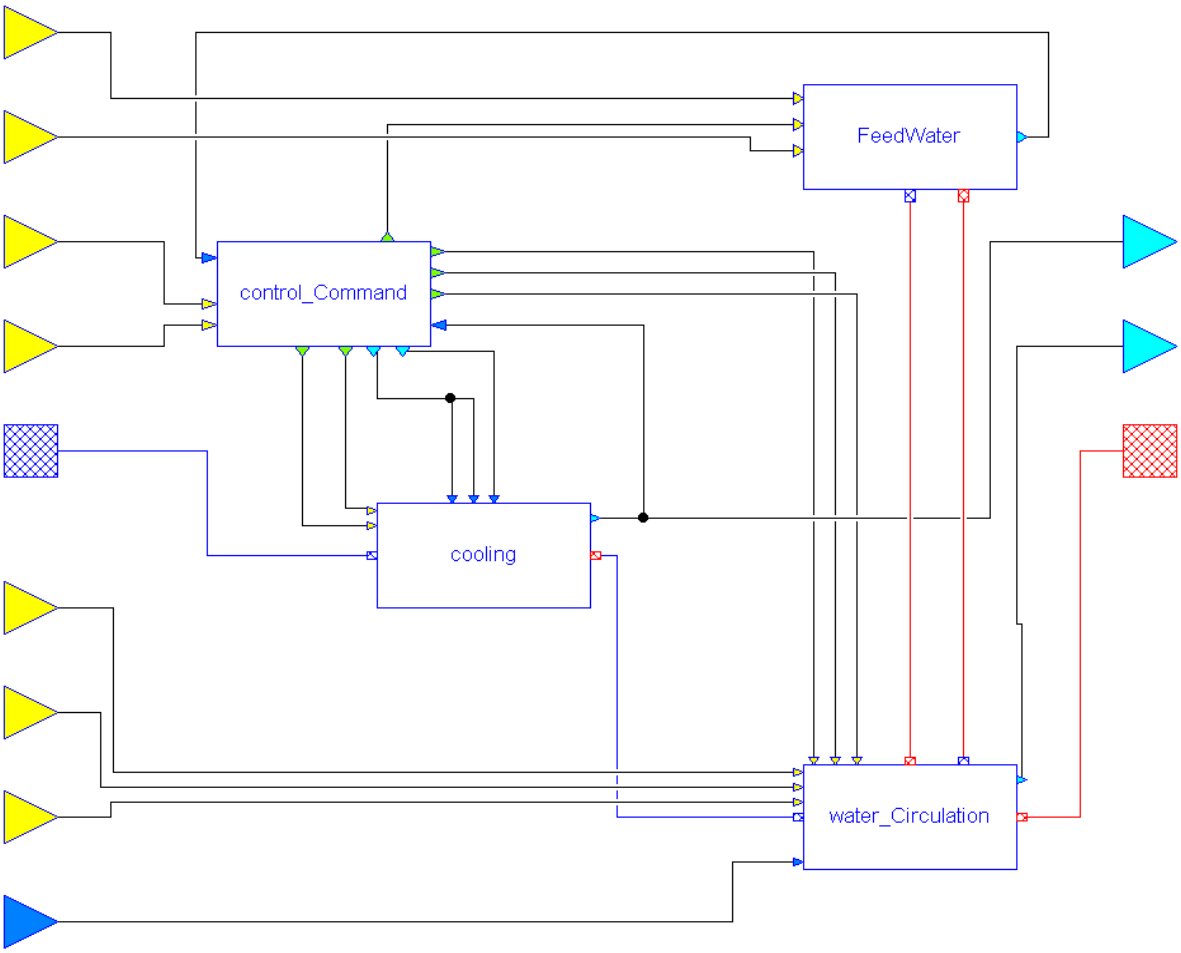


Figure 9 : sous-fonctions du modèle de comportement

La fonction contrôle-commande est elle-même subdivisée en sous-fonctions :

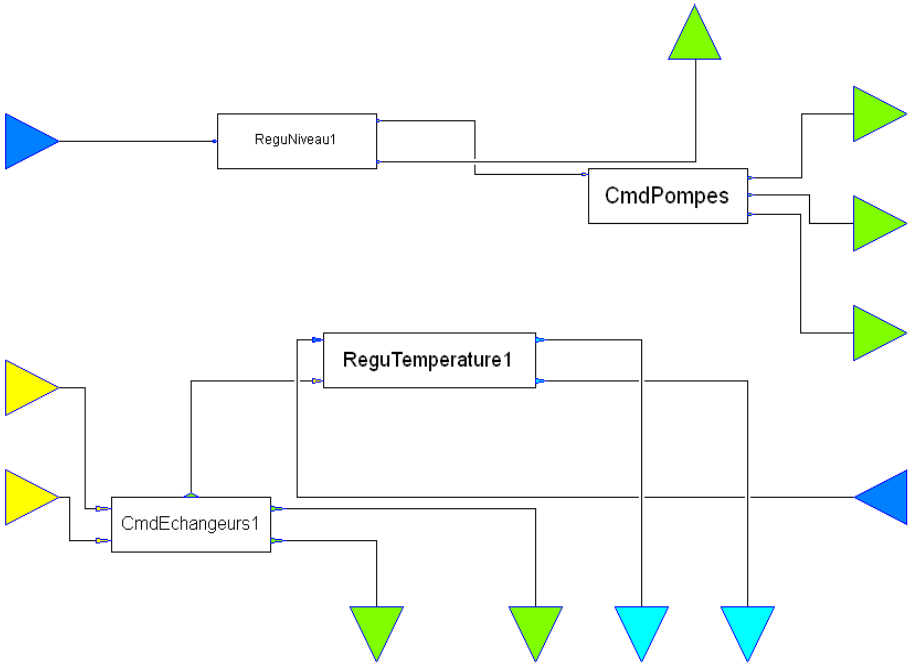


Figure 10 : sous-fonctions de la partie contrôle-commande

Le détail de l'implémentation de la fonction reprise en normal-secours des pompes (fonction logique CmdPompes) et de la fonction de régulation de température (fonction analogique ReguTemperature1) est présenté dans les deux figures suivantes :

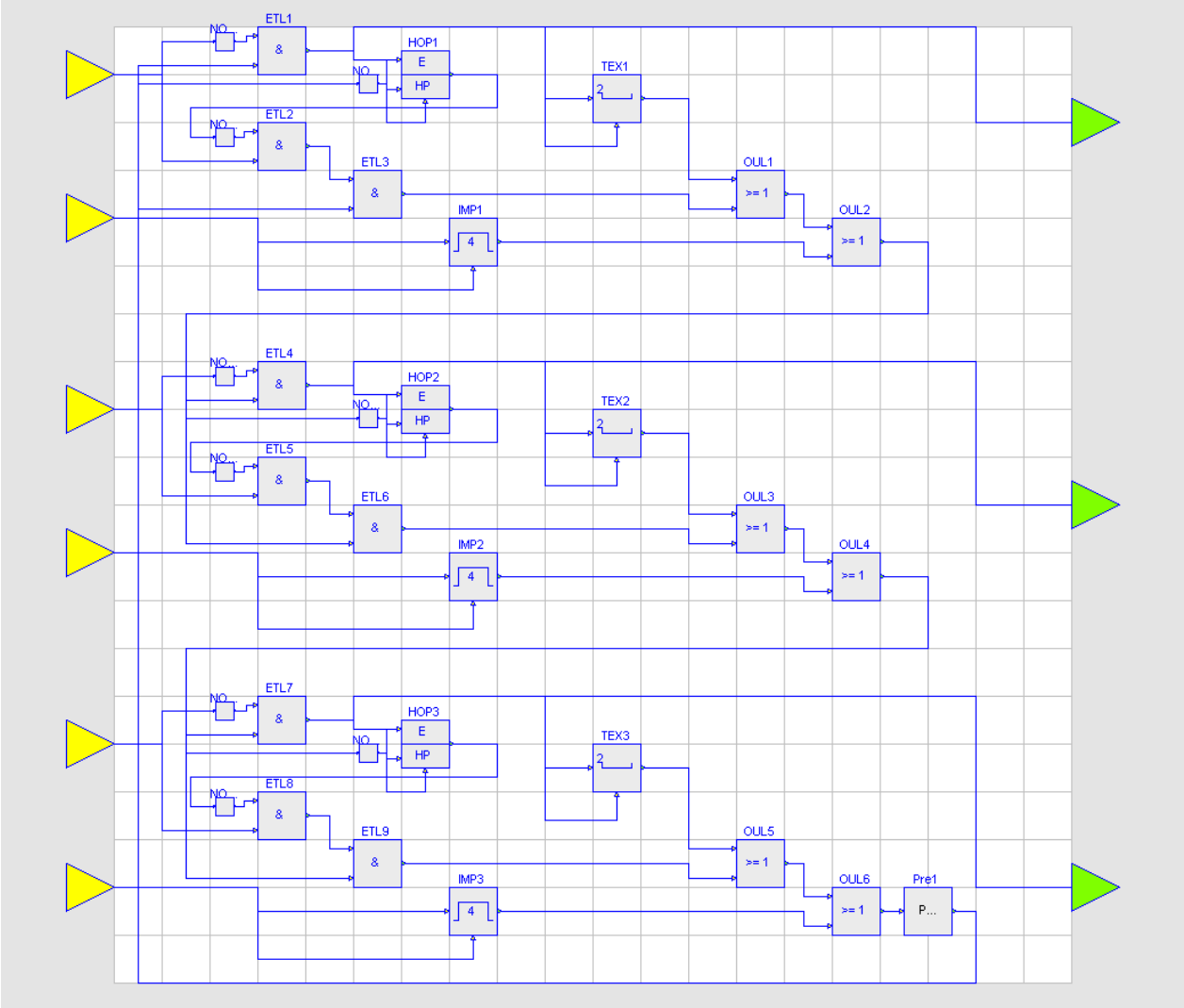


Figure 11 : Fonction de contrôle-commande de reprise en normal secours des pompes

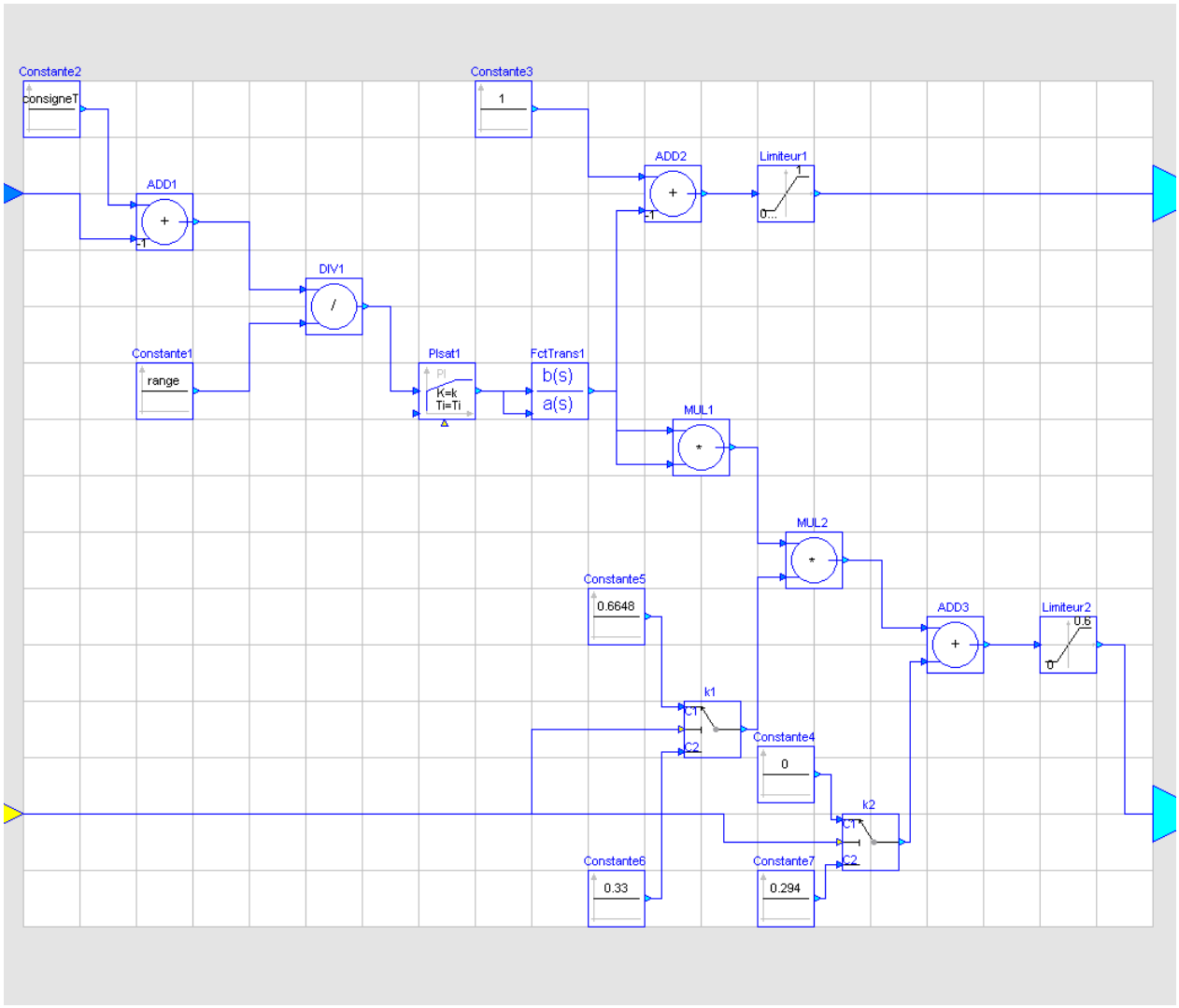


Figure 12 : Fonction de contrôle-commande de régulation de la température du circuit

4.4. Le superviseur

Le superviseur vient s'interconnecter avec les modèles de comportement, d'environnement et de propriétés par des ports de type multiplexeur. Le schéma ci-dessous montre le détail de cette structure.

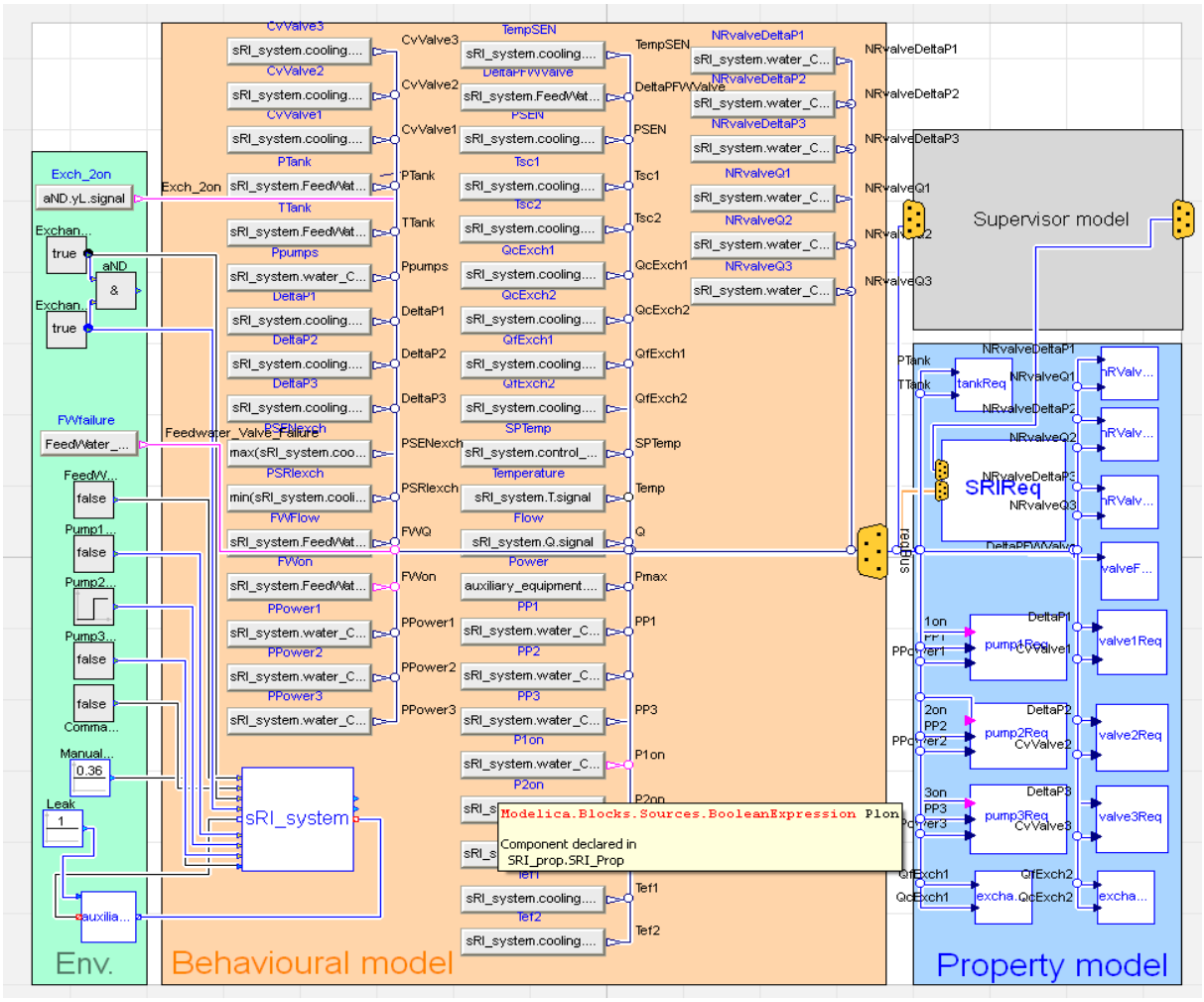


Figure 13 : interconnexions des modèles d'environnement, de comportement, de propriétés avec le superviseur

Le modèle de comportement est donc composé de machines à états pour les différents composants :

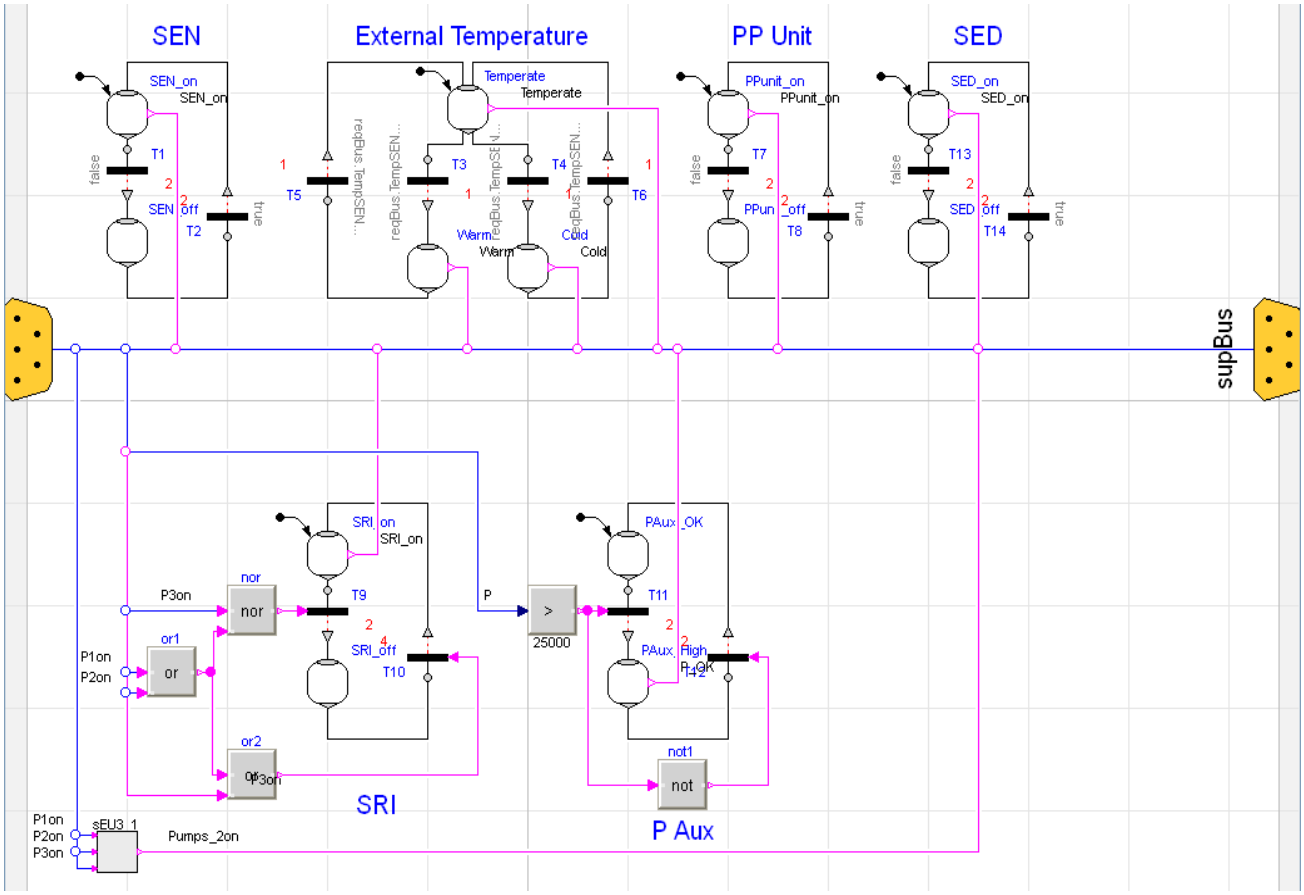


Figure 14 : machines à états composant le superviseur

Ces différents états concernent des parties du modèle de tailles différentes :

- Au niveau système : la machine à états « SRI » va identifier dans quel état se trouve le système étudié (on ou off) à partir des mesures venant du modèle de comportements.
- Au niveau modèle d'environnement : les circuits SEN qui sont en interface avec le système étudié sont également modélisés par une machine à état qui permet de les faire passer de l'état on à l'état off et inversement. Cela est géré par la définition des scénarios de tests.

4.5. Exploitation des modèles

Actuellement, les modèles sont initialisés par les études de fonctionnement qui permettent au travers de la mise au point du modèle de trouver le point de fonctionnement permettant de démarrer le modèle dans un état de fonctionnement normal. Cela intègre différents paramètres dont le choix de méthodes d'intégration, le cadence d'échantillonnage.

Le déroulement des scénarios de tests se fait en définissant un profil de stimuli au travers du modèle d'environnement. Les stimuli appliqués vont également influer sur le superviseur qui fera évoluer l'état du système et des composants en temps réel. Les scénarios de tests retenus pour l'instant correspondent à ceux définis dans les documents de référence (DSE et PPE), mais aucun calcul de couverture n'est effectué.

Par ailleurs, l'évaluation des propriétés est effectuée en temps réel au cours de la simulation. Les rectangles verts dans lesquels sont représentées les propriétés passent en rouge si une propriété n'est plus respectée. Si la propriété redevient valide, le rectangle repasse en vert mais sera encadré en rouge pour indiqué que la propriété n'a pas été respectée au moins une fois.

Une analyse des fichiers de résultats de tests pourra permettre d'expliquer pourquoi il y a eu non-respect de la propriété et ainsi d'initier une action de correction.

La génération de tests et l'analyse des résultats ne sont donc à ce stade pas automatisées. Ce cas test devrait permettre de par les modèles déjà développés et les fonctions des outils de modélisation support de tester la méthodologie qui sera développée par le projet VACSIM.

4.6. Le système PTR « Possibilité pour le Test de Remplissage et refroidissement de réservoirs »

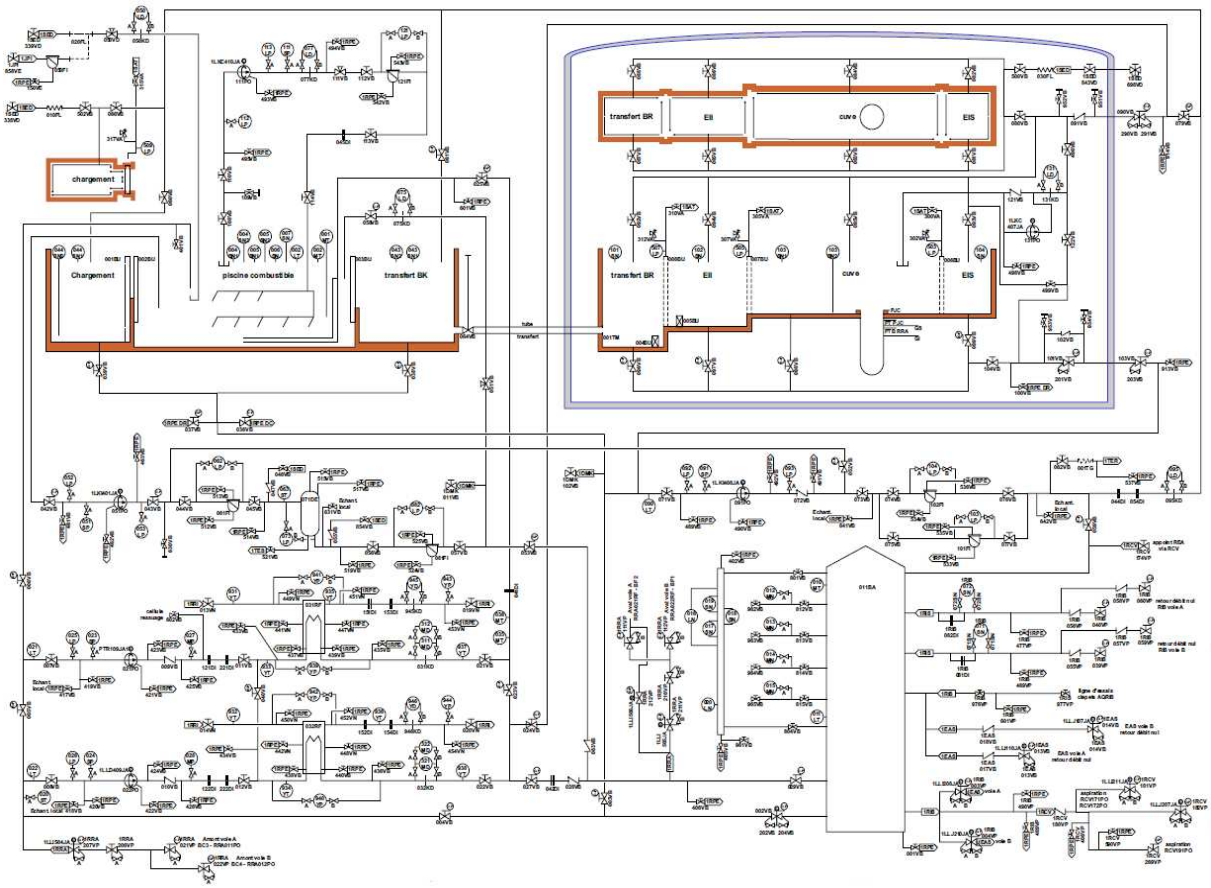


Figure 15 : Partie mécanique du système PTR

Des exemples de propriété à vérifier au niveau du système concernent le non-dépassement de seuils de température, l'absence de cavitation de pompes, l'absence de fuite lors de transfert d'eau, la non-sollicitation de soupapes ou le respect de configurations autorisées (circulation de fluide, isolement, mise en eau / remplissage ou en air / vidange).

Ce cas d'étude pourrait être utilisé pour un démonstrateur Dassault Systèmes d'un simulateur air / eau permettant d'exécuter les consignes normales de lignages des circuits.

5. Conclusion

La simulation ne doit pas remplacer la réflexion dans les études. Ce qui précède contribue à définir un environnement permettant de produire rapidement une grande quantité de cas de test par simulation et de vérifier simultanément de nombreuses propriétés, ce qui peut être utile dans le cas d'un test statistique d'un système critique. Les limites connues de la simulation sont d'abord celles de la modélisation, un phénomène non-représenté ne pouvant être étudié.

Les développements du lot 2 du projet VACSIM ne visent de ce fait pas à fournir un outil à utiliser en boîte noire dans la validation des systèmes. Il s'agit seulement d'investiguer la faisabilité d'un outil permettant de faciliter le travail des physiciens et ingénieurs responsables de procédés et de produits, par l'introduction de fonctions automatisant l'utilisation des simulateurs et l'analyse de leurs résultats.

Parmi les inconvénients de l'approche proposée, on note donc essentiellement la possibilité d'une utilisation d'un système automatisé sans examen critique de ses résultats. Concernant des systèmes critiques, cette difficulté peut être contournée par des définitions claires des conditions d'emploi des outils développés.

Les avantages escomptés de l'approche sont :

- La possibilité de produire facilement un grand nombre de cas d'étude, ce qui peut être exigé pour certains systèmes de contrôle-commande critiques ;
- La possibilité de compléter et de confirmer le résultat des études manuelles actuelles par un système automatisé produisant une étude supplémentaire à moindre effort ;
- La possibilité de faciliter les tests de non-régression des systèmes critiques partiellement modifiés, du fait d'une potentielle nouvelle production automatique de l'ensemble des cas de test.

6. Documents de référence

[1] Einsatz von Simulatoren zur virtuellen IBS der C16 HLT bei Modernisierungsprojekten am Beispiel RWE Power KW Neurath, Block D, *Dr. Heinz-Jürgen Wüllenweber, RWE Power AG, Grevenbroich, Jürgen Brunner, Siemens AG, Erlangen, Ludger Küppers, Kraftwerksschule e.V., Essen,* Conférence VGB « Kraftwerke im Wettbewerb 2011 » 29 et 31 mars 2011, Karlsruhe (D)

[2] Modern Nuclear DCS Hardware Testing Verification Using Simulation, *Jody Ryan, Pascal Gain, Corys,* 7th ANS International Topical Meeting on NPIC&HMIT, November 2010, Las Vegas, Nevada

[3] Test Selection Strategies for Lustre Descriptions in GATeL. *Bruno Marre, Benjamin Blanc*, MBT Workshop, ETAPS'04 Satellite Event, 2004.

[4] Critères de test et génération de séquences de tests pour des systèmes réactifs synchrones modélisés par des calculs flot de données et contrôlés par des automates étendus, *Christophe Junke,* Thèse de Doctorat de l'Ecole Centrale Paris, 9 janvier 2012.

[5] Autorité de Sûreté Nucléaire (ASN). Règle Fondamentale de Sûreté RFS-II.4.1.a du 15/05/2000 « Logiciels des systèmes électriques classés de sûreté »

[6] Norme Internationale CEI 60964 "Centrales nucléaires de puissance – Salles de commande – Conception" (2009)

[7] Projet Européen ITEA 2 'EuroSysLib', sous-projet (sWP) 7.1 « Properties Modeling » EDF R&D, Dassault Aviation, Dassault Systèmes, aout 2011

[8] Definition and Verification of the Control Loop Performance for Different Power Plant Types, *Nataliya Knierim-Dietz, Lutz Hanel, Joachim Lehner, Research Project Supported by VGB Power Tech*, Institute of Combustion and Power Plant Technology, Université de Stuttgart, mai 2012

[9] Building meaningful timed models of closed-loop DES for verification purposes, *Matthieu Perin, Jean-Marc Faure, LURPA ENS Cachan,* Control Engineering Practice, doi:10.1016/j.conengprac, 2012.05.002

[10] Closed-loop System Modeling, Validation, and Verification, *Sebastian Preusse, Hans-Christian Lapp, Hans-Michael Hanisch, University of Halle-Wittenberg*, 17th IEEE International Conference on Emerging Technologies and Factory Automation, septembre 2012

[11] A theory of timed automata. *Rajeev Alur, David L. Dill*. Theoretical Computer Science, 126(2) 183–235, 1994.

[12] AMT: A property-based monitoring tool for analog systems. *Nickovic, D., Maler, O.* In Formal Modeling and Analysis of Timed Systems. (2007) 304–319

[13] LARVA — safer monitoring of real-time java programs (tool paper). *Colombo, C., Pace, G.J., Schneider, G.* In: SEFM. (2009) 33–37

[14] Efficient Constraint-Based Dynamic Strategies For Generating Counterexamples, *Hélène Collavizza, Nguyen Le Vinh, Michel Rueher, Samuel Devulder, Thierry Gueguen.* 26th ACM Symposium On Applied Computing, Software Verification and Testing Track

[15] Constraint-Based BMC: A Backjumping Strategy, *Hélène Collavizza, Le Vinh Nguyen, Olivier Ponsini, Michel Rueher, Antoine Rollet.* STTT Journal, 2012. DOI: 10.1007/s10009-012-0258-6

[16] The Flasher Manager Benchmarks, *Hélène Collavizza, Le Vinh Nguyen, Olivier Ponsini, Michel Rueher, Antoine Rollet.* HAL : hal-00720921

7. Glossaire

Abréviation	Signification
DFL	Diagramme Fonctionnel Logique
HIL	Hardware-In-the-Loop. Les tests HIL réunissent une simulation de la partie opérative avec le système cible de contrôle-commande
IHM	Interface Homme Machine.
MIL	Model-In-the-Loop. Les tests MIL réunissent une simulation de la partie opérative avec la spécification de contrôle-commande
SIL	Software-In-the-Loop. Les tests SIL réunissent une simulation de la partie opérative avec le logiciel de l'application de contrôle-commande
SRI	Circuit de Réfrigération Intermédiaire
PTR	Circuit de Possibilité de Test de Remplissage et de Refroidissement de Réservoirs