3. INFORMATIQUE ET RÉSEAUX
COMPUTER SCIENCE AND NETWORKS

3.1 Modélisation hétérogène et logiciel enfoui
Heterogeneous modeling and embedded software

La conception d'un système fait appel à différents métiers qui utilisent chacun des concepts et des outils de modélisation spécifiques adaptés à l'aspect ou la partie du système qu'ils traitent, ainsi qu'à la phase de conception et au niveau d'abstraction auquel ils le traitent. La nécessité de modéliser la totalité du système de façon cohérente - afin de simuler son comportement, valider formellement certaines propriétés ou générer une implémentation pour une architecture cible - implique donc de raisonner sur un modèle hétérogène du système, ce qui oblige à définir précisément la sémantique des modèles métier et de leur combinaison.
La modélisation hétérogène permet donc de construire un modèle global d'un système (sujet 1) qui permet de le concevoir tout en prenant en compte des contraintes non fonctionnelles, et de le valider en raisonnant formellement sur ses propriétés (sujet 2).
Ces travaux s'inscrivent dans le cadre de l'ingénierie dirigée par les modèles et des recommandations de l'Object Management Group (OMG).

The design of a system involves several technical specialties that each have their own mindset and modeling tools. These tools suit the needs for a specific abstraction level, a particular aspect and a given part of the system, as well as a design phase. The need for a global and consistent model of the system - for simulation, formal proof of properties or code generation for a target platform - requires the ability to work on the system using a global heterogeneous model. Such a model is useful only if we are able to precisely define the semantics of each domain-specific model and of their combination in an heterogeneous model.
Heterogeneous modeling allows the construction of global models of a system (topic 1) which allows a designer to take non-functional constraints into account while building a system, and to validate the system using formal techniques (topic 2).
This work is done in the context of Model Driven Engineering and follows the recommendations of the OMG (Object Management Group).

Sujets

1. Modélisation et validation des systèmes

Conception de modèles d'exécution pour les systèmes hétérogènes.
Conception séparée du contrôle et des traitements pour faciliter les preuves.
Adaptation de la sémantique d'un modèle de calcul à son environnement d'exécution.
Conception de composants adaptatifs, réutilisables selon plusieurs modèles de calcul.
Modélisation des liens entre différentes vues d'un même système, vérification de la cohérence entre ces vues.
Génération de scénarios de test pour valider le comportement de systèmes hétérogènes.
Dans le cadre d'UML, formalisation de la sémantique des modèles de calcul ou d'exécution. Définition de méta-modèles et de profils, approche par transformation de modèles.

2. Validation formelle

Modularité des preuves et formalisation des mécanismes de sûreté pour permettre de composer des modules en conservant les propriétés de sûreté de chacun d'entre eux. Utilisation en particulier de l'approche réactive synchrone, de la méthode B et des réseaux de Petri.

Topics

1. Modeling and Validation of Systems

Design of execution models for heterogeneous systems.
Separate design of control and processing to make formal proofs easier.
Adapting the semantics of a model of computation to its runtime environment.
Design of adaptive components, reusable in different models of computation.
Modeling of the links between several views of a given system, consistency checking of these views.
Generation of test cases for validating the behavior of heterogeneous systems.
In the context of UML, formal definition of the semantics of models of computation. Definition of meta-models and profiles, transformations of models.

2. Formal proofs

Modular proofs and formal description of safety mechanisms to allow the composition of modules while preserving their individual safety properties. Use of the synchronous reactive approach, of the B method, and of Petri nets.

Pour tout renseignement s'adresser à : *For further information, please contact:*

Frédéric BOULANGER
Sujet 1 / *Topic 1*
Département Informatique
Campus de Gif
Tél. : 33 (0) 1 69 85 14 84
E-mail : frederic.boulanger@supelec.fr

Dominique MARCADET
Sujet 1 / *Topic 1*
Département Informatique
Campus de Gif
Tél. : 33 (0) 1 69 85 14 73
E-mail : dominique.marcadet@supelec.fr

Guy VIDAL-NAQUET
Sujet 2 / *Topic 2*
Département Informatique
Campus de Gif
Tél. : 33 (0) 1 69 85 14 75
E-mail : guy.vidal-naquet@supelec.fr

Verification of safety properties on heterogeneous systems

Vérification de propriétés de sûreté sur des sytèmes hétérogènes

Christophe Jacquet
Frédéric Boulanger
Dominique Marcadet

Résumé *Les systèmes industriels complexes sont en général composés de sous-systèmes décrits à l'aide de différentes méthodes de modélisation. Nos travaux visent à permettre aux concepteurs d'utiliser le formalisme de description le plus adapté à chaque sous-système, tout en étant capable d'exprimer le recollement entre les différents sous-systèmes. Dans cet exemple, nous montrons comment cette approche permet de valider les aspects critiques d'un système doté d'un contrôleur synchrone. Des propriétés de sûreté, exprimées par le concepteur sur les signaux du système, sont traduites automatiquement en observateurs du comportement du contrôleur. Les outils de model-checking permettent alors de vérifier formellement la conformité du système aux propriétés de sûreté spécifiées.*

Introduction

We describe a method for the design and code generation of heterogeneous software systems, which allows the specification of *safety properties* on the system and their formal verification. We distinguish two aspects in the design of a heterogeneous system: data processing, which produces the outputs from the inputs, and control, which determines the schedule and parameters of the operations. Data processing is described by data-flow models, whereas control is described by state machines or synchronous languages. Designing control independently from data processing allows the formal verification, re-use and independent modification of the parts of the system. We propose a modular approach, in which data processing is modeled by a data-flow network of *processing components*, and control is modeled by a unique *control component* whose inputs and outputs are pure events. Such a description of an application can be used both for deployment [1] and for validation [2]. System designers can express safety properties, which apply *globally* to the inputs and outputs of the application. Our main contribution is the automatic translation of these properties into *local* properties of the control component. The latter properties are transformed into observers in the formalism used for the controller. In this way, classical formal methods and tools can be used to check the control component directly: what is proved is what gets executed.

Application Description Language for Verification

The ADLV language allows one to describe an application based on components connected through events (event source, event sink) and data flows (data flow out, data flow in). Figure 1 is an ADLV description of a cruise control system.

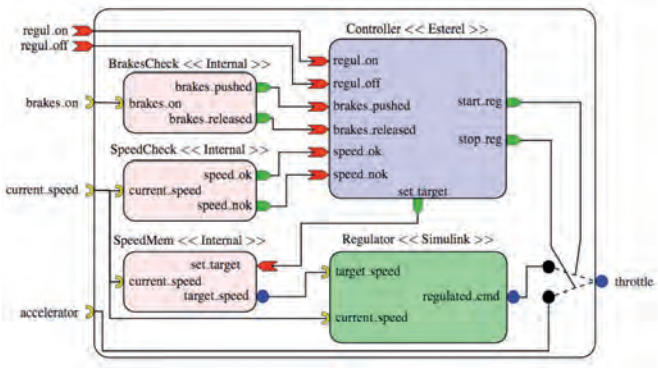

Figure 1: Cruise control in ADLV.

Components can be of three types:

- the controller of the application, written in Esterel or Lustre,
- external components modeled using various methods, for instance Simulink components or C code. They are viewed as black boxes,
- internal components, which translate between events and data flows and are described in ADLV. Figure 2 shows the description of the SpeedCheck internal component in ADLV's textual syntax.

```
internal component SpeedCheck {
        sink FloatFlow current_speed;

        publishes PureEvent speed_ok {
                when current_speed >= 40
                  && current_speed <= 130;
        }
                                        publishes PureEventspeed_nok {
                                                when current_speed < 40
                                                  || current_speed > 130;
                                        }
};
```

Figure 2: Internal component SpeedCheck.

Verification of Safety Properties

We allow the designer to express safety properties (LTL formulae of the form $\Box f$ where $\Box$ is the "always" temporal operator, and f contains past operators only) that involve any of the application's signals. For instance, the following formula expresses the fact that the cruise control should not set the target speed when the current speed is off-limits:

$$\Box\neg(\textbf{(speed < 40 } \vee \textbf{ speed > 130)} \wedge \textbf{set\_speed)}$$

From that, we automatically create an *intermediate formula* which refers to controller events only. Therefore, we translate data flow signals or other events into *intervals* between two events. For instance, from the definition of the *SpeedCheck* component, one sees that the sub-formula **speed < 40 ∨ speed > 130** starts to be true when event **speed_nok** is emitted, and ceases to be true when **speed_ok** is emitted. This means that this sub-formula corresponds to the interval **[speed_nok, speed_ok[**. The safety formula above yields the following that involves only controller events and can be translated into a Lustre or Esterel observer:

$$\Box\neg(\textbf{ [speed\_nok, speed\_ok[} \wedge \textbf{ [set\_speed, !set\_speed[ )}$$

Approximate Observers

It may not be possible to find exact matches between safety properties and the ADLV description. For instance, if we replace **speed > 130** by **speed > 140** in the safety formula above, it is no longer possible to generate an observer with the same method. However, we can generate an *approximate* observer since **(speed > 140)** ⇒ **(speed > 130)**. This means that the event which corresponds to **speed > 130** happens *before* that triggering **speed > 140**. Therefore the interval **[speed_nok, speed_ok[** is an *approximation* of the interval corresponding to the new formula, but it is *too narrow*. When we use it to generate an observer, if model-checking detects a failure, this will be a real failure of the system, but the check may miss some misbehaviors. Conversely, with too large an interval, an observer that passes would guarantee that the system conforms to the safety property, however, that observer might detect some false positives.

Conclusions

Thanks to the method presented above, the designer of a system may express safety properties in the global context of the system, and yet can use model-checking tools to get an exhaustive diagnosis on the system controller. In cases when an *exact* observer cannot be generated, the method provides a clear indication of whether a sub- or over-verification is performed, allowing the designer to get useful results even in this case. The method has been implemented in an extensible tool chain that currently targets Esterel and Lustre.

This work has been performed in the context of the Usine Logicielle project (www.usine-logicielle.org) and is partially financed by the System@tic Paris-Région Competitiveness Cluster (www.systematic-paris-region.org).

References

[1] A. Bouzoualegh, D. Marcadet, F. Boulanger and C. Jacquet, "An Architecture Description Language for Verification in Component-Based Software", Proceedings of the 32nd Annual IEEE International Computer Software and Applications Conference, COMPSAC 2008, Jul 2008. pp. 365-368.

[2] C. Jacquet, F. Boulanger and D. Marcadet. "From Data to Events: Checking Properties on the Control of a System", Memocode 2008, the sixth ACM/IEEE International Conference on Formal Methods and Models for Co-Design, Jun 2008. pp. 17-26.

3.2 Détection d'intrusions
Intrusion Detection

Les organisations humaines de toutes tailles dépendant aujourd'hui très fortement de leurs systèmes informatiques, si bien que leur sécurité est devenue un enjeu crucial. Il convient donc, préventivement, de définir et de mettre en œuvre des politiques de sécurité. En outre, en tenant compte de failles toujours possibles, il convient également, en seconde ligne de défense, de contrôler le respect de la dite politique. En d'autres termes, on cherchera à détecter toute action non conforme à la politique. C'est la détection d'intrusions. Les principaux outils de détection d'intrusions couramment utilisés de nos jours ne permettent de détecter que des attaques déjà répertoriées (approche dite par signature). Ils se heurtent donc au problème quotidien des nouvelles formes d'attaques. Pour contourner ce problème, la « détection d'anomalies » consiste à confronter le comportement courant de l'entité surveillée à un modèle de comportement de référence construit préalablement. Classiquement, le modèle de référence est construit par apprentissage. Par exemple, le comportement d'un processus applicatif est observé pendant le temps nécessaire à la découverte de toutes les suites d'une longueur donnée d'appels systèmes émis par ce processus. En phase de détection, toute suite de cette longueur qui ne se trouvent pas dans le modèle donne lieu à alarme. En effet, une attaque par injection de code, par exemple peut expliquer cette situation.

Computer and network security is nowadays a major concern, as human organizations of any kind and any size depend on its information system. The answer essentially lies in the capacity to define and enforce security policies; subsequently it is important to consider that flaws are always possible and to monitor systems in order to detect possible exploitation of these flaws. This is intrusion detection. Intrusion detection systems currently used can only detect already known attacks (misuse detection). Thus, they face the problem of daily appearing new form of attacks. To avoid this problem, "anomaly detection" aims at comparing a current observed behaviour of the monitored entity, to a reference model previously built. Generally the reference model is built through a learning process. For example, an applicative process is, in an initial learning phase, observed during the period of time needed to identify all the possible system call sequences of a given length. Then, in a detection phase, sequences occurring but not present in the identified possible sequences lead to the emission of an alert. Indeed, such a situation may be the result of a code injection attack.

Sujets

Notre contribution au domaine de la détection d'intrusions s'inscrit dans le cadre de la détection d'anomalies. Cependant, nous cherchons à construire des modèles de référence en évitant toute forme d'apprentissage. En effet, un modèle appris peut être incorrect (la phase d'apprentissage peut inclure des attaques) ou incomplet (il est difficile voire impossible de savoir si toute les situations possibles ont été rencontrées pendant l'apprentissage). Pour éviter l'apprentissage, nous avons exploré diverses pistes. Par exemple, dans l'exemple présenté ci-contre, le modèle de référence est implicite, chaque version constituant un modèle pour les autres. Une autre approche qui nous semble particulièrement intéressante, consiste à faire de la politique de sécurité la référence. Nous avons proposé un mécanisme de surveillance des flux d'information paramétré par la politique, qui permet de détecter tout accès à l'information illégal au regard de la dite politique, et ce même lorsque l'information n'est plus dans son conteneur d'origine. Ce mécanisme de détection de flux illégaux peut être implanté à divers niveaux. Nous disposons d'une implantation au niveau système d'exploitation et au niveau langage (java/JVM).

Topics

Our contributions to the intrusion detection field are mainly related to anomaly detection. Nevertheless, we avoid, as far as possible, relying on a learning mechanism. Indeed, learning may lead to incorrect (attacks may occurs during the learning phase) or incomplete (it is difficult if not impossible to know if all the possible behaviours have been seen during the learning phase) models. To avoid learning, we have explored several research tracks. For example, the work presented here after proposes to consider an implicit reference model. Here, the user requests are forwarded to different modules that implements the same functionality but through diverse designs. Any difference between responses that are returned by these modules can be interpreted as a possible corruption of one or several modules. This provides a way to detect intrusions in the diversified system. Another appealing approach lies for us in policy-based intrusion detection. Here, the detection system is aware of the security policy that constitutes actually the reference model. We have proposed an approach to monitor information flows allowing detecting violation of a security policy even if the information is no more in its original container. Such an approach can be implemented at various levels. We have an implementation for a Linux operating system and another one for the Java Virtual Machine.

Pour tout renseignement s'adresser à :

Ludovic MÉ
Équipe SSIR
Campus de Rennes
Tél. : 33 (0) 2 99 84 45 00
E-mail : ludovic.me@supelec.fr

For further information, please contact:

Anomaly detection through design diversity
Détection d'anomalies par diversité fonctionnelle

Ludovic Mé
Eric Totel

Résumé *Les approches de détection d'intrusions dites « comportementales » (« anomaly detection », en anglais) demandent qu'un modèle de comportement de l'entité surveillée soit explicitement défini. Nous proposons ici une nouvelle approche, basée sur la diversification fonctionnelle, qui permet de contourner cette contrainte en proposant un modèle de référence implicite. Nous avons expérimenté, sur une architecture à trois serveurs, deux algorithmes de détection. Le premier (boîte blanche) compare les sorties des serveurs, tandis que le second (boîte grise) compare les flux d'information internes aux serveurs. Ces deux algorithmes donnent des résultats expérimentaux intéressants. Le second offre cependant un taux de couverture plus large et, de surcroit, permet une forme de diagnostic de l'anomalie détecter.*

Design diversity aims at fault tolerance by performing a function in two or more versions and then by executing a comparison algorithm (difference detection) on the different results. To greatly reduce the probability of common-mode failure in the different versions, they are independently designed and produced, by different teams using different tools. Design diversity is of course a very expensive approach, as the same software has to be developed several times. However, many of the services available via the Internet are already implemented as Components-Off-The-Shelf (COTS) on a wide range of operating systems. We have here a "natural" diversity of these services, as they offer the same functionalities. That is why we proposed to build on COTS diversity.

Albeit two COTS implementing the same service should theoretically follow the same specification, there is unfortunately no proof that it is the case. Actually, it is only true for the COTS user interfaces, that are explicitly provided, for instance by some international standard. The comparison algorithm can obviously only be applied on the outputs that are defined by the common specification, and not on other outputs that may be defined by a COTS specific specification part.

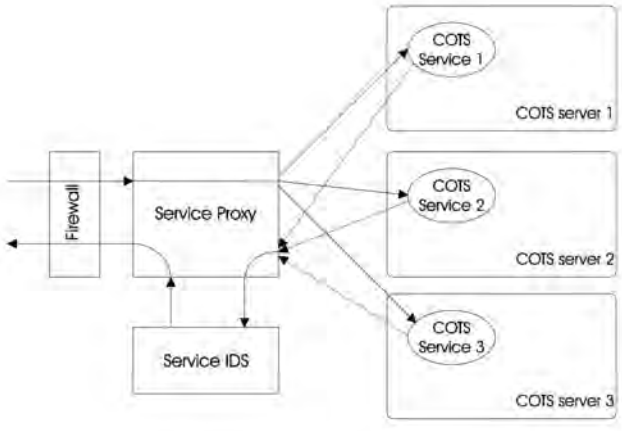

Figure 1: General architecture

The architecture we proposed (Fig. 1) is composed of three components: a proxy, an IDS (Intrusion Detection System), and a set of servers. The role of the proxy is to handle the client requests. It forwards the request from the client to the COTS servers and then forwards the response from the IDS to the client. It is the sole part of the architecture accessible directly to the clients but it is simple enough to be considered as secure. The IDS is in charge of comparing the responses from the COTS servers. If it detects some differences, it raises an alarm and it informs the proxy of the response that has been elected by a voting algorithm. This algorithm is in charge of choosing which response must be sent back to the client. A set of COTS servers provides the service requested by the client. These servers offer the same services but they are diverse in terms of application software, operating systems and hardware. This aims to reduce the probability of a common-mode failure: in the context of our studies, it aims at ensuring the vulnerabilities of the different servers are not correlated and thus that an intrusion occurs in only one COTS server at a time. Because the other COTS servers are not affected by the same vulnerability, the architecture allows to detect the intrusion and even to tolerate it (1).

The output differences that can be detected are due to design differences either in the specific parts of the specifications, or in the part of the program covered by the common specification. We intend to detect differences that are the consequences of the exploit of vulnerabilities (these vulnerabilities are design faults and can be part of any of the two classes that have been listed above). The output differences detected that are due to classical design faults or specification differences would actually lead to false alarms and must be eliminated. In order to avoid known difference of servers' behavior, masking functions are applied to modify the request before it is processed or the response after processing. In both cases, these differences are experimentally identified.

In order to compare the responses from the COTS servers, we have proposed and evaluated two mechanisms: a black box approach that consist in comparing the outputs of the diversified services (http responses) without any knowledge of the internals of the servers [1], and a grey box approach that relies on an intrusive observation of the activities that occur on the diversified servers (information flow graphs generated by the activities on the servers) [2].

Our experiments have shown that both approaches can provide a high coverage of detection and a low level of false positives. However, the black-box approach cannot obviously detect intrusions that have no impact on the network outputs. The grey-box approach increases thus the detection coverage. Moreover, it adds an interesting diagnosis capability to the IDS, as the analysis of the differences between the information flow graphs enlightens the effects of the intrusions at the OS level. The advantage here is thus to propose to the administrator more than a simple intrusion detection mechanism: it brings him an evidence of the intrusion and its causes. As far as we know, this is the first anomaly detector that offers such a capability.

(1) A study of the vulnerabilities of IIS and Apache proves that there are very few common mode failures between them.

References

[1] F. Majorczyk, E. Totel, and L. Mé. "COTS Diversity Based Intrusion Detection and Application to Web Servers". In proceedings of the 8th RAID Symposium. Springer Verlag, LNCS 3858, September 2005.

[2] F. Majorczyk, E. Totel, L. Mé, and A. Saidane. "Anomaly Detection with Diagnosis in Diversified Systems using Information Flow Graphs". In proceedings of the 23rd IFIP SEC Conference. September 2008.

3.3 Sécurité des réseaux auto-organisés
Self-organized Networks Security

Nous regroupons sous la dénomination « réseaux auto-organisés » tout réseau se caractérisant par l'absence d'une autorité globale en charge de définir et maintenir d'une part son infrastructure, d'autre part la politique de sécurité qui s'y applique. Cette absence d'autorité se traduit pas la nécessité d'intégrer, au niveau de chaque nœud, des mécanismes permettant de constituer et de gérer le dit réseau. La notion de réseaux auto-organisés est naturellement présente dans l'informatique ambiante (ubiquitous computing) avec les réseaux ad hoc (ou MANET pour Mobile Ad hoc NETwork), mais également dans les systèmes distribués avec les réseaux pair-à-pair (P2P). Dans le cas des réseaux ad hoc, la problématique est la mise en place des mécanismes de routage nécessaires à l'interconnexion des nœuds. Dans le cas des réseaux P2P, les problèmes relèvent davantage du partage et de la distribution de l'information. Du point de vue de la sécurité, les réseaux auto-organisés n'introduisent pas réellement de nouveaux problèmes. En revanche, l'absence d'autorité centrale en charge de gérer les infrastructures et les usages a des conséquences importantes. En particulier, chaque nœud doit considérer qu'il évolue dans un environnement « à risque » et donc mettre en place les mécanismes lui permettant de se protéger contre les nœuds malveillants, éventuellement en collaborant avec les nœuds qu'il sait être bienveillants.

We call « Self-Organized Networks » any network that has no global authority in charge of defining and managing its infrastructure as well as its security policy. The absence of such an authority leads to the necessity to integrate, on each node of the network, the mechanisms and services that are mandatory to build and manage the network. The notion of self-organized networks is present in the ubiquitous computing with the ad hoc network or MANET (Mobile Ad Hoc NETwork) and in the distributed systems with the P2P networks. In the case of ad hoc networks, the main problem is to establish the routing infrastructure that allows interconnecting the nodes. In the context of the P2P networks, the challenge is to propose a distributed mechanism to share and manage the information of the nodes. From the security point of view, the self-organized networks do not really introduce new problems. However, the absence of a central authority in charge of managing the infrastructures and the services has serious consequences. In particular, each node have to consider that it evolves in an insecure environment, and so has to implement its own mechanisms to enforce its security against malicious nodes, eventually by collaborating with some trusted nodes.

Sujets

1. Sécurité des réseaux ad hoc
Expression formelle des règles de confiance implicite.
Détection des incohérences entre les informations de routage.
Détection des nœuds ayant un comportement malicieux.

2. Sécurité des réseaux P2P
Système de contrôle d'accès distribué à base de cryptographie à seuil adaptatif.
Protection contre les attaques de type Sybil.
Détection et révocation de nœuds ayant un comportement malicieux.

Topics

1. Security of ad hoc networks
Formal specification of the implicit trust relations.
Detection of inconsistencies of the routing information.
Detection of nodes having malicious behavior.

2. Security of P2P networks
Distributed access control based on a adaptive threshold cryptography scheme.
Protection against Sybil attacks.
Detection and revocation of nodes having malicious behavior.

Pour tout renseignement s'adresser à :

Christophe BIDAN
Équipe SSIR
Campus de Rennes
Tél. : 33 (0) 2 99 84 45 00
E-mail : christophe.bidan@supelec.fr

For further information, please contact:

Trust management as a security solution for OLSR protocol
Gestion de la confiance au service de la sécurité du protocole OLSR

Christophe Bidan

Résumé *La notion de confiance, quoique implicite, est toujours présente dans le fonctionnement des protocoles, en particulier, entre les entités qui participent aux opérations de routage. Dans ce travail, nous nous sommes intéressés à la gestion de la confiance (trust management) comme une solution de sécurité pour le protocole OLSR (Optimized Link State Routing Protocol). Cette approche s'adapte particulièrement à la nature mobile, distribuée et auto-organisée des réseaux ad-hoc. De plus, la gestion explicite de la confiance permet aux entités de raisonner avec et à propos de la confiance, les rendant ainsi plus robustes pour la prise de décisions concernant les autres entités.*

Context

Several research studies were conducted during the last few years aiming at developing protocols for networks whose nodes communicate directly with each other to relay messages without the support of a central entity. This operating mode characterizes the ad hoc networks, for which the Internet Engineering Task Force (IETF) standardized some routing protocols such as the Optimized Link State Routing Protocol (OLSR, RFC3626).

Due to the absence of a fixed network infrastructure as well as a central entity, traditional security solutions are not adapted to the ad hoc networks. Especially, since all nodes are involved in the routing protocol, each node should consider other nodes as potential attackers. Thus, new security solutions have been proposed to secure the routing protocol in ad hoc networks. These solutions are often focused on the security of the routing information (in terms of confidentiality and integrity), and are based on either a central authority, or the cooperation between the nodes. However, they do not take into account the nodes that exploit vulnerabilities in the routing protocol.

The basic idea of our work is that the notion of trust, although almost implicitly, is always present in network protocols. Thus, it can be regarded as an unreasonable behavior that protocol entities do not explicitly take trust into account. Conversely, managing trust explicitly allows the entities to reason with and based on trust, which in turn helps these entities to take decisions regarding the other entities. Such an approach fits particularly with the characteristics of ad-hoc networks, since each node can individually take decisions about its neighbor nodes.

Contribution

We have focused on the OLSR (Optimized Link State Routing) protocol. First, we have analyzed the OLSR protocol to demonstrate that, according to its specifications, the establishment of the routing structure is associated to a process based on trust rules through cooperation among nodes for discovering neighbors, selecting routers and announcing topology information. We have formalized these implicit trust rules, and shown that for many attacks against the OLSR protocol, the attacker generally misbehaves with respect to these trust rules.

Based on these trust rules, we have proposed for OLSR the integration of semantics checking and trust reasoning into each node, so as to allow a self-organized control to help nodes to detect attacks. The solution consists in verifying the consistency between the information received by a node. By using this reasoning, and without modifying the bare OLSR protocole, each node can evaluate the behavior of its neighbor nodes, and detect misbehavior nodes, and so decided whether it can trust or not these nodes.

We have used the GlomoSim Simulator and the OLSR patch developed by the Niigata University to simulate the attacks and the mistrust-based detection process. We have added to this patch a module implementing mistrust rules, so as to allow the simulations to be carried out for the cases of the bare OLSR protocol and the OLSR with trust reasoning. Simulation results have demonstrated the effectiveness of the verification based on trust reasoning in the attack detection. The results allow setting up verifications that each node can perform to assess the correct behavior of the other nodes and detect attacks against OLSR. It is important to mention that the OLSR protocol (messages) is unchanged, and the detection is based on local observation of each node.

Finally, we have proposed two kinds of counter-measure. Some of the vulnerabilities of the OLSR protocol are due to the easy usurpation of node's identity or to the lack of links verification during the neighborhood discovery step. They can be solved by a preventive solution based on message signature. Other vulnerabilities cannot be mitigated by preventive measures. We have then proposed a corrective misbehavior and inconsistency detection solution that permits to isolate the malicious nodes and to share information about detected attacks so as to make the malicious nodes known of all the nodes of the network.

Conclusion

In this work, we have demonstrated that trust based reasoning can be used to enforce the security of the OLSR protocol. This result has motivated us to apply the same approach on other ad hoc routing protocols. We have begun to study the implicit trust rules of the AODV protocol. Moreover, we also plan to use the cooperation for the propagation of mistrust, in order to enforce a reputation system by verifying trust relationships before cooperating with the other nodes.

References

[1] F. Majorczyk, E. Totel, and L. Mé. “COTS Diversity Based Intrusion Detection and Application to Web Servers”. In proceedings of the 8th RAID Symposium. Springer Verlag, LNCS 3858, September 2005.

[2] F. Majorczyk, E. Totel, L. Mé, and A. Saidane. “Anomaly Detection with Diagnosis in Diversified Systems using Information Flow Graphs”. In proceedings of the 23rd IFIP SEC Conference. September 2008.

3.4 Systèmes d'informations hétérogènes et adaptatifs
Heterogeneous and Adaptive Information Systems

Les systèmes d'informations actuels sont caractérisés par leur volume croissant et leur hétérogénéité en termes de domaines, de sources, de représentation et de structuration des informations. L'accès à une information pertinente et adaptée aux utilisateurs dans ces systèmes est un vrai challenge. Les besoins des utilisateurs sont difficiles à traiter, d'une part, parce qu'ils ne sont pas toujours formulés explicitement, et d'autre part, parce qu'ils sont évolutifs.
Les systèmes visés sont les systèmes de recommandation, les hypermédias adaptatifs et les systèmes de recherche d'informations adaptatifs.
Nos travaux portent sur la définition des modèles des utilisateurs, des contextes, des services et des ressources, ainsi que sur l'exploitation de ces modèles pour adapter le fonctionnement des systèmes d'informations aux utilisateurs.

The current information systems are characterized by the increasing number and the heterogeneity of available resources. These resources are heterogeneous on different points of view: domains, sources, representation and structure. The access to a relevant and adapted information for each user in these systems is a challenge. The user needs are difficult to deal, on one hand, because they are not formulated explicitly and, on the other hand, because they are evolutive.
The aimed systems are recommendation systems, adaptive hypermedia and more generally adaptive information systems.
Our aim is to define models of the users, models of the contexts, models of the services and models of the resources and to exploit these models in order to adapt information systems to the user needs.

Sujets

1. Systèmes de recommandation
Conception de systèmes basés sur :
- des modèles utilisateurs hybrides intégrant différentes sources d'informations (numériques ou symboliques) ;
- des méthodes d'apprentissage pour déterminer les recommandations adaptées à l'utilisateur.

2. Web adaptatif
Modèles, méthodes et architectures permettant, l'adaptation d'hypermédias à l'utilisateur (profil et objectif) ainsi qu'au contexte d'utilisation. Construction d'une plateforme générique pour le développement d'applications adaptables. La généricité est réalisée grâce à l'utilisation de la logique pour la partie moteur d'exécution, et des langages du Web Sémantique pour la représentation des connaissances.

3. Recherche d'information adaptative
Modélisation et conception de systèmes prenant en compte le profil, les préférences les interactions et l'environnement de l'utilisateur.
Spécification de modèles de fonction de correspondance utilisant des modèles utilisateurs et les caractéristiques des corpus de documents traités.

4. Extraction de connaissances
Classification Conceptuelle de données hétérogènes basée sur les treillis de Galois.
Développement d'un système d'intégration et de recherche d'informations dans des ressources semi-structurées, guidé par les connaissances du domaine (ontologie).
Apprentissage de documents multimédia annotés pour l'adaptation à l'utilisateur.

Topics

1. Recommendation Systems
Conception of systems based on:
- *hybrid user models which take into account different information sources (numeric and symbolic data);*
- *learning methods for adapting recommendations according to user needs.*

2. Adaptive Web
Models, methods and architectures allowing, on the fly, the adaptation of hypermedia systems to the user (profile and aims) and to the context. Creation of a generic platform for the development of adaptive applications. The genericity is achieved through the use of logic for the execution machine and the use of the Semantic Web languages for knowledge representation.

3. Adaptive Information Retrieval
Modelling and conception of systems considering the profile, preferences, interactions and environment of the user in order to personalize the answers.
Specification of models of matching functions using user models and features of corpora of documents.

4. Knowledge Extraction
Conceptual clustering of heterogeneous data by Galois lattices.
Ontology-based system for integration of semi-structured resources. The purpose is to allow users to access directly relevant parts of documents as answers to their queries.
Learning methods on annotated multimedia documents for user adaptation.

Pour tout renseignement s'adresser à : *For further information, please contact:*

Géraldine POLAILLON
Nacéra BENNACER
Sujets 1, 4 / *Topics 1, 4*
Département Informatique
Campus de Gif
Tél. : 33 (0) 1 69 85 14 78/14 71
E-mail : prenom.nom@supelec.fr

Yolaine BOURDA
Sujet 2 / *Topic 2*
Département Informatique
Campus de Gif
Tél. : 33 (0) 1 69 85 14 80
E-mail : yolaine.bourda@supelec.fr

Bich-Liên DOAN
Sujet 3 / *Topic 3*
Département Informatique
Campus de Gif
Tél. : 33 (0) 1 69 85 12 56
E-mail : bich-lien.doan@supelec.fr

Reusing Adaptive Hypermedia Models
Réutilisation de modèles d'hypermédias adaptatifs

Nadjet Zemirline
Yolaine Bourda

Résumé *La conception d'Hypermédias Adaptatifs (HA) est une tâche longue et difficile qui peut-être facilitée par la réutilisation de systèmes génériques. Nous avons proposé une plateforme générique (GLAM [1]) pour les HA qui permet par spécialisation de ses modèles de résoudre ce problème. Nous proposons ici, un processus semi-automatique, basé sur des patrons et des règles de déduction, de spécialisation de modèles. Celui-ci a été implémenté comme un plug-in de la plateforme Protégé et testé dans le cadre des HA.*

Creating Adaptive Hypermedia Systems

Nowadays, there is a growing demand for personalization and the "one-size-fits-all" approach for hypermedia systems is no longer applicable. Adaptive hypermedia (AH) systems adapt their behavior to the needs of individual users. Thus, adaptive hypermedia systems are tools to access information based upon the user's profile represented in a user's model. They also require a domain model to represent the application domain knowledge. These two kinds of models may be expressed in an AH-specific language or a standard language (RDF, OWL). Adaptation mechanisms, either rule or trigger based, which are needed in adaptive hypermedia rely on these models.

The creation of an adaptive hypermedia system is too often made from scratch and the re-use of existing models (user or domain) is very rare although more and more annotated resources are available. But, if a user wants to use a specific AH system, he needs to translate his models into the specific format understood by the system and to use the vocabulary specific to that system. Furthermore, she also needs to translate all the instantiations of her models (i.e. the resources and their metadata). This task is tedious and time-consuming and we want to avoid it. Our objective is to allow the creator of an adaptive hypermedia to reuse his models (his vocabulary) and his models' instantiations without any change of format or vocabulary.

We are currently working on the GLAM (Generic Layered Adaptation Model) [1] platform defined for an entire class of adaptive hypermedia systems. The platform is made of a generic adaptation model relying on generic user and domain models. Specific systems can be obtained by specializing the GLAM generic user and domain models. However, this specialization process is not always easy to perform and must be supported to make the design process easier and faster. We aim to automate this process which has been so far entirely manual. The proposed approach relies on W3C standards (OWL, SWRL) widely used.

Main aspects of the approach

Given two models, a generic model belonging to the GLAM platform and a specific model provided by a particular AH creator, we propose an approach to support the construction of a model that would integrate all the particularities of the specific model and be usable by the GLAM adaptation engine. In the approach, mappings must be defined between elements of both models and then validated at the structural level. Our approach relies on the AH creator who has a very good understanding of his model. He will be responsible for semantic validation while all the structural verifications will be done automatically by our system. The main steps of the approach are the following:

1. Specification, by the AH creator, of equivalence and specialization mappings between classes of the generic and the specific models, merging the whole generic GLAM model and the mapped classes of the specific model (together with the associated mapping links) in order to obtain a new model (cf. (1) Fig. 1).
2. Automatic computation of additional mappings between classes, the mappings and the linked classes being added in the being built model (cf. (2) Fig. 1).

Starting from the mappings between classes specified by the AH creator, other mappings can be automatically deduced. We propose to adopt a pattern-based approach to achieve this deduction. Pattern-based approaches for mapping identification across models assume that structural regularities always characterize the same kind of relations. The idea is to deduce the nature of the relation R between a class of the specific model and a class of the generic model

3. Automatic computation of mappings between elements different from classes. Checking consistency of the new model created by the merging process (cf. (3) Fig. 1).

To do so, our system uses structural knowledge applicable to whatever the model is (user or domain model). As models are expressed in OWL, structural knowledge has been modelled in a meta-model based on the OWL meta-model. Inferences on knowledge modelled in the meta-model are performed using SWRL rules.

4. Validation by the AH creator of the deductions made by the system in step 3. (cf. (4) Fig. 1).

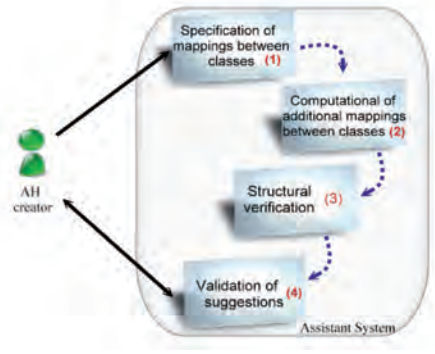

Figure 1: The diagram of the architecture of our assistant system

Related Works

There are several approaches for performing a semantic integration depending on the degree of integration usually referred to as ontology mapping, aligning or merging. These approaches are based either on instances of the two given ontologies that are to be mapped (bottom-up) or on concepts (top-down). None of them have been used to merge abstract models with specialized ones. Here, we focus on this specific point. The models to be merged are relatively small. The merging process is performed once at the design time. Generic models are composed of abstract classes which have no instances. The designer of the system knows the models to be integrated in the system very well and can then provide simple correspondences between their elements.

Conclusions

We have proposed a solution enabling the user to create an adaptive hypermedia with the GLAM system re-using his own models and consequently his own resources and their metadata. Furthermore, this approach is generic and consequently usable whatever the application domain is. We have implemented it as a Protégé plug-in (MESAM) and made some experiments.

This work has been done in collaboration with Chantal Reynaud (Université Paris-Sud XI, CNRS (LRI) & INRIA - Saclay Ile-de-France / Projet Gemo).

References

[1] Jacquiot, C., Bourda, Y,. Popineau, F., Delteil, A., Reynaud. C.: GLAM: A generic layered adaptation model for adaptive hypermedia systems. In: 4th International AH2006, Springer, pp. 131-140. Springer, Heidelberg, Allemagne (2006).

[2] Zemirline N., Reynaud C., Bourda Y., Popineau F.: A Pattern and Rule-Based Approach for Reusing Adaptive Hypermedia Creator's Models. In: 16th EKAW, PP. 17-31, Springer, Catania, Italy (2008).

3.5 Systèmes situés
Situated Systems

L'approche située du traitement de l'information vise à développer des machines autonomes capables d'évoluer dans des environnements naturels et donc hautement non-stationnaires (du fait de la présence d'humains par exemple) et possédant des facultés d'adaptation. En particulier, l'approche située de la perception considère la captation (ou la perception), l'analyse (ou le raisonnement) et la décision (ou l'action, le comportement) comme différentes modalités d'un même processus visant à l'accomplissement d'une tâche dans un environnement. La perception n'est alors plus considérée comme passive comme dans les approches ascendantes classiques ou les informations sont traitées de manière hiérarchisée, mais plutôt comme faisant partie d'une démarche active, par interaction, visant la découverte d'informations pertinentes pour la tâche à laquelle le système est dévolu. C'est donc la tâche et l'interaction avec l'environnement qui pilotent l'extraction d'information et pas uniquement une connaissance a priori du concepteur, ce qui confère aux machines adaptabilité et autonomie.

The situated approach to information processing aims at developing autonomous machines able to evolve in natural environments and so highly non-stationary (because of the presence of humans, for instance) and showing adaptation capabilities. Especially, the situated approach to perception considers sensing (or perception), analysis (or reasoning)and decision (or action, behavior) as different modalities of a same process which goal is the achievement of a task in a given environment. Perception is not considered anymore as passive as in standard approaches where information is processed in a hierarchical manner. It is regarded as being an active process happening through interaction with the environment so as to collect relevant information regarding the application at seek. Information extraction is therefore driven by the task and interactions with the environment and not only according to designer's knowledge and expertise. This makes machines adaptive and more autonomous.

Sujets

1. Environnements intelligents
Les moyens de captation et de traitement d'information peuvent maintenant être enfouis dans les objets du quotidien et dans nos environnements (domicile, bureau, voiture). Ces moyens peuvent être utilisés pour améliorer la qualité de vie de tout un chacun mais peuvent aussi permettre le maintien à domicile de personnes dépendantes ou en situation de handicap. Néanmoins, c'est l'humain qui doit être au centre des développements technologiques et c'est pour cette raison que l'adaptation doit être intrinsèque aux méthodes développées dans ce cadre.

2. Robotique cognitive
L'approche située ne peut se concevoir que s'il y a interaction physique entre les machines et le monde. La robotique est un des meilleurs moyens d'ancrer l'intelligence dans le monde réel et en devient une application privilégiée de l'approche située.

3. Interfaces homme-machine
L'interaction est au cœur de l'approche présentée et la nécessité d'adaptation autonome qui est recherchée est souvent créée par la présence d'humains dans l'environnement des machines. Les interfaces entre machines et humains peuvent prendre plusieurs formes comme les interfaces vocales, multimodales ou cerveau-machine.

4. Modèles d'inspiration biologique
Des méthodes d'apprentissage d'inspiration biologique sont mises au point comme la modélisation corticale ou l'apprentissage par renforcement.

Topics

1. Smart Environments
Sensors and information processing units can now be embedded into day-to-day life objects as well as in our environments (home, office, cars). Those means can be used to enhance the quality of life for each of us but can also enable elderly, impaired or dependent persons to stay longer at home. However, the human has to be at the centre technological developments. This is why adaptation should be at the heart of the methods developed in this framework.

2. Cognitive Robotics
The situated approach can only be envisioned if physical interactions between machines and the real world happen. Robotics is one of the best means to anchor intelligence in the real world. It therefore becomes a privileged application of the situated approach.

3. Human-Machine Interaction
Interaction is at the heart of the situated approach and the necessity of autonomous adaptation is often created because of the presence of humans in the machines environment. Interfaces between man and machines can be of several types such as voice-based, multimodal or brain-machine interfaces.

4. Bio-inspired models
Artificial learning methods inspired from biology are under focus such as cortical modeling and reinforcement learning.

Pour tout renseignement s'adresser à : ***For further information, please contact:***

Olivier PIETQUIN
Équipe IMS
Campus de Metz
Tél. : 33 (0) 3 87 76 47 70
E-mail : olivier.pietquin@supelec.fr

Hervé FREZZA-BUET
Équipe IMS
Campus de Metz
Tél. : 33 (0) 3 87 76 47 35
E-mail : herve.frezza-buet@supelec.fr

Dynamic neural fields
Champs neuronaux dynamiques

Hervé Frezza - Buet

Résumé *Les champs neuronaux dynamiques sont une population d'unités de calcul qui modélisent la surface corticale. Les connexions forment une topologie 2D qui sert de base aux compétitions au sein de cette population. Les calculs unitaires sont décrits par une équation différentielle en espace et en temps, faisant de l'ensemble de la population un système dynamique complexe. La population répond à un profile de stimulation en contrastant ce profil, le ramenant à quelques activités isolées. Cette opération est utilisée comme processus de décision dans nos systèmes situés, pouvant réduire l'information tout en restant fortement lié à l'évolution des entrées.*

An equation for competition

Dynamic neural fields are defined by a population of computation al units, referenced in a continuous space X of positions (usually 1D or 2D). At each position x in X, a unit is fed with an input $i(x)$ and computes $u(x)$ as an output. This output is driven by a differential equation, as the one presented hereafter, proposed by Amari [1]. Lateral coupling weigh tkernel w is usually a difference of Gaussians.

$$\frac{du(x,t)}{dt} = -u(x,t) + \int_{x'} w(|x-x'|) f\big(u(x,t)\big) dx' + i(x,t) + h$$

Once discretized according to both space and time, the equation is the activation rule of a cellular automata, which is a discrete dynamical system. This system continuously evolves toward some equilibrium configuration. If parameters are chosen in a suitable way, the $u(x)$ distribution at equilibrium is made of sparse bumps of activity, placed where the input distribution $i(x)$ is the most compact. If $i(x)$ represents the result of some filtering process centered at x, the field activity consists of selecting over the field X the few places where the filtering process is the mostly responding (cf. figure 1).

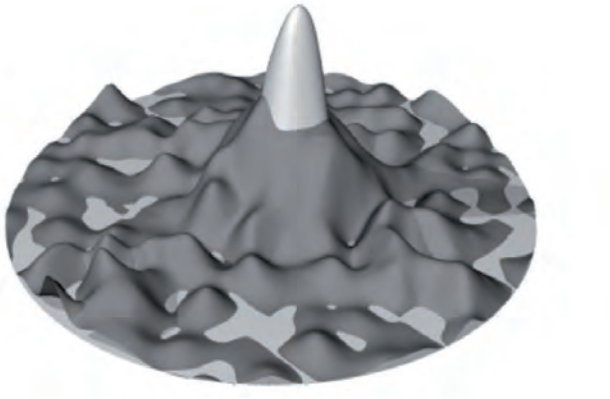

Figure 1: Neural field (here a disk). The light surface represents $i(x)$ and the darker one represents $u(x)$, that is the distributed decision computed by the field. Such a decision process is the basis for the situated decision mechanisms investigated in our team.

This selection can be viewed as a robust reduction mechanism, reducing information according to a population compromise rather than an arbitrary threshold. This allows us to consider decisions, that usually belong to the symbolic domain due to their binary nature, as processes that are kept grounded in the analogical real world. This grounding ensures that the decision is permanently recomputed as input profile changes.

Multimodal self organization

On the basis of such a decision process, we have set up an unsupervised learning technique, close to Kohonen's Self-Organizing Maps (SOM). As opposed to SOMs, one advantage here is that the learning process can be parallelized since competition is distributed. This has allowed the implementation on our PC cluster (InterCell project [2]). From this learning module, a multimodal adaptive architecture as been defined [3] allowing to built controllers. It is made of initially undifferentiated elements (the fields), that become more and more specialized as the system interacts with its environment to actually achieve the control. Handling such complex dynamical systems requires analysis and visualizing tools, that are provided for cluster simulation in the framework of InterCell, allowing us to address large scale problems. Moreover, as reinforcement learning is also investigated, new equations for dynamic neural fields have also been proposed [4].

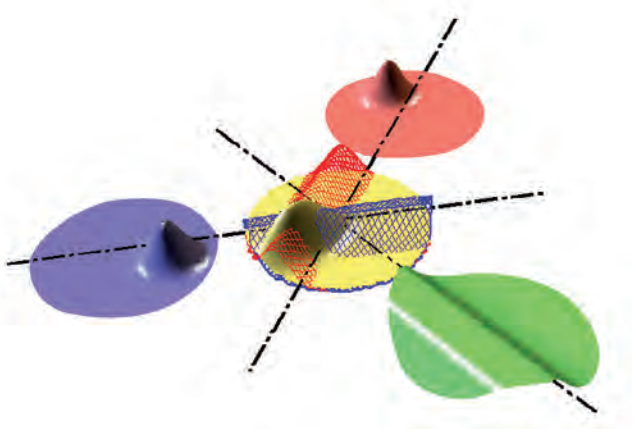

Figure 2: Multimodal architecture based on 4 neural fields. Each field deals with a specific modality, or is at the intersection of some other fields, to handle multimodal computation. Each field is also performing competition, and all competitive processes are coupled through inter-field connections. Once learning is achieved, filters are tuned so that related ones stand actually connected places.

Conclusion

Dynamic neural fields can be used to set up a decision process without requiring the designer to make any rule explicit. This is coherent with our main goal to propose new algorithms for situated perception. Such mechanisms are inspired from the dynamics of cortical computation, as described in biology. Indeed, the cortex in mammals is one of the major adaptive structures for setting up multimodal behaviors.

References

[1] S-I Amari, Biological Cybernetics, 27:77–87, 1977

[2] http://intercell.metz.supelec.fr

[3] O. Ménard, H. Frezza-Buet, "Model of multi-modal cortical processing: Coherent learning in self-organizing modules". In Neural Networks, 18(5-6):646-655, 2005

[4] L. Alecu, H. Frezza-Buet, "Reconciling neural fields to selforganization". In ESANN'09, 2009.

3.6 Systèmes distribués et grilles de calculs
Distributed Systems and Computing grids

Les architectures multi-cœurs sont devenues courantes et progressent rapidement, les clusters sont également devenus des moyens de calcul très répandus. D'autre part, la France a récemment intensifié sa stratégie d'équipement en supercalculateurs et l'Europe dispose maintenant de grilles de calculs scientifiques opérationnelles. Toutes ces architectures parallèles et distribuées requièrent des algorithmes et des paradigmes de programmation différents, mais se retrouvent aujourd'hui combinées dans des « machines hétérogènes », comme des clusters de machines multi-cœurs, ou des supercalculateurs équipés d'accélérateurs matériels. Nos recherches visent à concevoir des algorithmes et des environnements de développements pour des architectures hétérogènes à large échelle. Cette démarche inclut la conception de mécanismes de tolérance aux pannes, problème incontournable dans les systèmes parallèles à large échelle.
Nous concevons également des architectures de services distribués permettant de profiter pleinement et facilement de ressources informatiques réparties sur plusieurs sites. Enfin, nous analysons autant les performances calculatoires qu'énergétiques de nos systèmes distribués.

Multi-core architectures are available and improving, and clusters are now well known architectures installed in many laboratories and companies. Moreover, today France purchases more supercomputers and Europe has operational scientific computing Grids. These architectures requires different parallel programming strategies, and are now mixed in "heterogeneous machines", like clusters of multi-core nodes, or supercomputers with hardware accelerators. We aim to design algorithms and development environments for large scale heterogeneous architectures. This approach includes the design of fault tolerance mechanisms, as failures are unavoidable on large scale systems.
We also design service oriented architectures, in order to easily use computing resources distributed on different sites. Finally, we study both computing and energetic performances of all our parallel and distributed systems.

Sujets

1. Distribution de calculs de contrôle stochastique en grande dimension
Nous avons distribué sur clusters de PCs et sur supercalculateurs IBM Blue Gene (32000 cœurs) des calculs de contrôle stochastique en grande dimension, appliqués à l'optimisation de problèmes de gestion de production d'énergie.

2. Distribution de calculs financiers sur clusters de GPUs
Les clusters de PC équipés d'accélérateurs matériels, comme les GPUs, permettent d'atteindre de hautes performances avec une faible consommation énergétique, mais nécessitent de concevoir des algorithmes parallèles complexes. Nous l'avons fait avec succès pour des pricings d'options européennes exotiques.

3. Mécanisme de tolérance aux pannes pour des applications sur cluster de PCs
Conception d'un modèle et d'un mécanisme de tolérance aux pannes au niveau applicatif, suivant des patrons de conception simples à utiliser et limitant les surcoûts d'exécution.

Topics

1. Parallelization and distribution of large scale stochastic control computations
We have distributed large scale stochastic control computations on large PC clusters and IBM Blue Gene supercomputers (32000 cores), to solve some optimisation problems of energy production management.

2. Distribution of financial computations on GPU clusters
PC clusters with GPU (on each node) achieve high performances with low energy consumption, but they require very complex parallel algorithms. We succeeded to do it for some exotic european option pricing.

3. Fault tolerance mechanisms for applications running on PC clusters
Design fault tolerance model and mechanisms, to use at application level accordingly to design patterns, and exhibiting very limited overheads.

Pour tout renseignement s'adresser à : *For further information, please contact:*

Stéphane VIALLE
Équipe IMS
Campus de Metz
Tél. : 33 (0) 3 87 76 47 20
E-mail : stephane.vialle@supelec.fr

Virginie GALTIER
Équipe IMS
Campus de Metz
Tél. : 33 (0) 3 87 76 47 36
E-mail : virgine.galtier@supelec.fr

Algorithmes multi-paradigmes pour des clusters de CPU et GPU multi-coeurs

Multi-paradigm algorithms for multi-core CPU and GPU clusters

Stéphane Vialle

Résumé *Les architectures parallèles modernes sont hétérogènes, comme des clusters de CPUs multicoeurs ou de GPUs. Ce type d'architecture peut atteindre des performances très élevées, mais nécessite des parallélisations complexes multi-grains et multiparadigmes. Les principaux défis à relever consistent à concevoir des algorithmes multi-paradigmes efficaces et des environnements de développement de haut niveau, et à maîtriser la consommation énergétique de ces architectures.*

Multi-paradigm programming of heterogeneous architectures

PC cluster are cheap to purchase, and today they are automatically multi-core PC clusters. Moreover it is easy to add a GPU card in each PC to get a GPU cluster, i.e: a PC cluster with "hardware accelerator" inside each node. However, this kind of heterogeneous architectures remain difficult to program. Their programming includes two level of parallelism, with different grains.

The first level is "coarse grained" and aims to distribute computations across the nodes, while minimizing inter-node communications and achieving these communications in parallel of the computations (overlapping computations and communications). The programming paradigm of this parallelism level is message passing, and the MPI library is the most common development tool for this paradigm. The second level of parallelism is "medium grained" and exploits the cores of a node. Its natural programming paradigm is memory sharing, and the most common associated programming tools are multi-thread libraries. They can be explicit multithread libraries, like POSIX threads, or they can supply a higher level of multithreading, like OpenMP based on pre-processing directives, or like Intel-TBB based on C++ templates. Finally, a third level of parallelism is "fine grained" and exploits SIMD computing units of GPUs, or SSE units of CPUs. Its programming paradigm is data parallelism, running a same operation flow on a vector of data. Standard are emerging, but currently programs for hardware accelerators are not portable. Beyond these three levels of parallelism it exist a "very coarse grained" level when using a multisite Grid of computing resources, interconnected through Wide Area Networks. The associated programming paradigm remains the Remote Procedure Call or the message passing, with asynchronous calls in order to hide the long communication times.

For heterogeneous architectures, we need to exploit simultaneously different grains of parallelism: to design multi-grain parallel algorithms and to use different parallel programming paradigms. From a pure technical point of view, it is possible to use both message passing with MPI, and multi-threading with OpenMP or data-parallelism of GPUs with CUDA. But the design of efficient multi-paradigm algorithms remains complex. We lack of algorithmic knowledge, and we lack of adapted high level development environment to reduce development times.

Algorithmic development examples

Recently we have designed two multi-paradigm parallel applications: an optimization of energy management, designed with EDF company on a multi-core CPU cluster and on a supercomputer, and exotic european option pricing, designed with ENPC and CERMICS laboratories on a GPU cluster.

The application of the optimization of the energy management uses complex stochastic control algorithms. Their distribution on coarse grained architectures requires redistributing data and computations at each iteration. Moreover, the communication scheme depends on previous computed values and computing nodes have to compute and to establish their routage plan at each iteration before to execute it. On each node, a second and medium grained parallelization allows to easily spread the local computations on the different cores of each CPU. Finally, despite many parallelism management operations and their associated overheads, our parallel algorithm and implementation (using MPI and OpenMP) have been efficient on a 256 dual-core PC cluster and on an IBM BlueGene/P with up to 8192 quad-core nodes (more than 32000 cores) [1]. An implementation on a GPU cluster is ongoing.

The parallelization of an exotic european option pricer on a GPU cluster appeared to be symmetric to the previously described parallelization of energy management optimizations. The coarse grained parallelization on a PC cluster has been straightforward to develop, leading to an embarrassingly parallel implementation. It does not require inter-node communications, excepted at the beginning and at the end of the application, in order to distribute data and to collect results. But the fine grained parallelization on GPU has required strong development efforts to be efficient. When calling GPU routines from a CPU routine, the most time consuming operations can be the data transfers between CPU and GPU memories. It is advised to transfer data onto the GPU card and to run the maximum number of operations before to transfer back results onto the CPU. In the case of the exotic european option pricing we have implemented several random number generators on the GPU, in order to run all computations on the GPU and to get uncorrelated random number suites. Finally, the parallelization on a 16 GPU cluster appeared 2.8 times faster than the parallelization on a 256 dual-core CPU cluster, and has consumed 28 times less energy! The product of the speedup and the energy saving showed it was 80 times more interesting to use our 16 GPU cluster instead of our 256 dual-core CPU cluster [2].

Perspectives

We are currently designing an algorithmic and a programming knowledge on heterogeneous architectures. However, we consider it is mandatory to design some modern development environments, adapted to these architectures and hiding the exact node features. Some of our research works, achieved in collaboration with INRIA and with EDF R&D, address this issue.

Acknowledgment

The author wants to thank EDF for supporting this research.

References

[1] S-P. Vezolle, S.Vialle and X. Warin. Large Scale Experiment and Optimization of a Distributed Stochastic Control Algorithm. Application to Energy Management Problems. Workshop on Large-Scale Parallel Processing (LSPP 2009). 2009.

[2] L.A. Abbas-Turki, S. Vialle, B. Lapeyre and P. Mercier. High Dimensional Pricing of Exotic European Contracts on a GPU Cluster, and Comparison to a CPU Cluster. Second Workshop on Parallel and Distributed Computing in Finance (PDCoF 2009). 2009.

3.7 Performance de systèmes
System performance

Notre objectif consiste à étudier un système de sa définition fonctionnelle à la proposition de solutions validées avec des performances évaluées. Nous nous intéressons donc : 1) au choix et à la validation de modèles calculables qui tiennent compte de tous les éléments pertinents du système, 2) à la définition formelle du problème étudié dans ce modèle et au calcul de sa complexité algorithmique, 3) à la proposition de solutions algorithmiques, centralisées ou distribuées selon les cas, notamment des solutions heuristiques dans le cas où le problème s'avère NP-complet, 4) à la validation de l'efficacité de ces algorithmes en termes de temps d'exécution et de qualité par leur introduction dans le modèle déjà construit et 5) à la résolution ce modèle modifié afin de décider si les heuristiques proposées ont un impact positif sur l'efficacité du système.
Pour pouvoir valider l'efficacité de nos solutions heuristiques, nous utilisons différents concepts et outils comme la théorie des files d'attente. Il existe un éventail de méthodes analytiques et empiriques (simulations) qui, en trouvant la solution du modèle, donnent des informations sur l'efficacité du système.

We study a system starting from its functional definition up to proposed solutions which are validated by evaluation of the system performance. We are interested in: 1) choice and validation of calculable models taking into account pertinent elements of the system, 2) formal definition of a studied system within the given model with its complexity computation, 3) proposition of algorithmic solutions (centralised or distributed), notably heuristic ones in the case of the NP-complete problem treatment, 4) validation of performance of these algorithms in terms of execution time and quality by their introduction into models which have been already constructed, and 5) resolution of the modified model in order to decide whether the proposed heuristic algorithms have a positive impact upon the system performance. To answer these questions, we use different concepts and tools such as the queueing theory. There is a wide range of analytic methods (Markov chains) and empirical ones (simulations) which, by finding the solution to the model, offer information about the system performance.

Sujets

1. Algorithmes distribués pour le routage multi-contraint satisfaisant la qualité de service dans un réseau inter-domaine
Algorithmes de réservation multi-contraintes dans un environnement BGP. Algorithmes de détection de congestion dans ce réseau. Validation par simulation.

2. Méthodes décompositionnelles pour les files d'attente
Matrices de Markov. Réseaux d'automates stochastiques. Algèbre de processus pour l'évaluation de performances.

3. Complexité et approximabilité de problèmes
Traitement de problèmes NP-complets. Problèmes de Steiner. Preuves d'inapproximabilité.

4. Performance et QoS des services IP multimedia dans les réseaux sans fil
UMTS, HSDPA, WiMAX, interaction entre TCP et lien sans fil, algorithmes d'ordonnancement, analyse du trafic dans des réseaux wireless (QoS, délai, gigue).

Topics

1. Distributed algorithms for multi-constraint routing satisfying Quality of Service in an inter-domain network
Algorithms for multi-constraint reservation in a BGP environment. Algorithms for congestion detection in the same environment. Validation by simulation.

2. Decompositional methods for queueing models
Construction of Markov matrices. Stochastic automata networks. Performance evaluation process algebra.

3. Problems complexity and approximability
Treatement of NP-complete problems. Steiner problems. Inapproximability proofs.

4. Performance and QoS of IP multimedia services in wireless networks
UMTS, HSDPA, WiMAX, interaction between TCP and wireless link, scheduling study, streaming traffic analysis in wireless systems (QoS, delay, jitter).

Pour tout renseignement s'adresser à : *For further information, please contact:*

Joanna TOMASIK
Sujets 1 à 3 / *Topics 1 to 3*
Département Informatique
Campus de Gif
Tél. : 33 (0) 1 69 85 14 79
E-mail : joanna.tomasik@supelec.fr

Mohamad ASSAAD
Sujet 4 / *Topic 4*
Département Télécommunications
Campus de Gif
Tél. : 33 (0) 1 69 85 14 43
E-mail : mohamad.assaad@supelec.fr

Exploiting the inter-domain hierarchy for the QoS routing
Exploitation de la hiérarchie inter-domaine pour le routage avec QoS

Joanna Tomasik
Marc-Antoine Weisser

Résumé *Le routage inter-domaine dans l'Internet est assuré par le protocole BGP qui garantit à chaque domaine l'indépendance des choix et des annonces des routes. Ce mécanisme seul ne permet pas la satisfaction de QoS et l'ingénierie de trafic. Pour pallier ces manques, nos travaux dans le réseau inter-domaine nous ont amené à exploiter la hiérarchie introduite par les relations commerciales existantes entre les domaines. Nous proposons un mécanisme envoyant des messages d'alerte pour communiquer l'état de congestion d'un domaine uniquement à chaque domaine concerné. Notre solution limite le nombre d'alertes envoyées grâce à la structure hiérarchique de l'Internet. Notre heuristique est distribuée et traite un problème NP-complet et inapproximable. Pour pouvoir valider notre mécanisme d'envoi d'alertes pour signaler la congestion possible, nous avons proposé un générateur de topologies aléatoires avec hiérarchie SHIIP (Supélec Hierarchy Inter-domain Inducting Program).*

Hierarchy of inter-domain network

An inter-domain network is composed of independent domains administrated by operators. Links connecting domains are characterized by two types of relationships: *P2C* and *P2P*. A *P2C* relationship links providers which sell connectivity to their customers. A *P2P* relationship exists between two domains which share connectivity. The inter-domain hierarchy has an impact on the current inter-domain routing because routes are established according to commercial relationships. The only routes present in an inter-domain network are composed of an uphill and a downhill component. A downhill component is a list of consecutive links labeled with a *P2C* relationship. An uphill component is a list of consecutive links labeled with a *C2P* relationship which is dual to a *P2C*. The uphill and downhill components are connected either by zero or by one *P2P* relationship. Such routes are called *valley-free.*

Problem modeling

Given a network, a traffic matrix and a routing matrix, we say that a node is *perturbed* if the total amount of traffic transiting through it, emitted by it, and sending to it is greater than its capacity. A *perturbed path* is a path containing at least one perturbed node. Given a network and a traffic matrix, our problem is to find a valley-free routing matrix which minimizes 1) the number of perturbed nodes *(network approach)*; 2) the volume of the traffic passing along perturbed paths *(traffic approach)*. We focus on the network approach only because we assume that minimizing the number of perturbed nodes may be a good heuristic approach to minimize the number of perturbed paths as well as the volume of perturbed traffic. In [2] we have proved that this problem is NP-complete and inapproximable. The proof consists to reduce the directed Steiner tree problem to our problem.

Alert sending algorithm

Our distributed algorithm is based upon the principle of sending alert messages which carry information about the domain congestion state. Each node informs its clients when it becomes perturbed in order to allow them to change their routing. Each node also keeps its neighbors informed when it returns to an operational state. We limit the range of their diffusion by taking advantage of the inter-domain hierarchy. Each node contains a list of possible next hops towards every destination. These lists are constructed with routes announced by BGP. Their construction guarantees that all next hops satisfy the valley-free property of routes. Each node can be in either of two distinct states: *green* or *red*. A node state is *red* if at least one of the two following conditions is satisfied: 1) the amount of traffic transiting through the node, sending from it and sending to it, is greater than its capacity; 2) at least one of its next hops which is a provider is in *red* state. A node cannot become *red* because of its customer and peer congestion. Because of this fact, our algorithm avoids the spreading of *red* nodes in the entire network when a single node becomes *red*. We use the hierarchy to limit the number of nodes in *red* state as well as the messages sent: a node which has a customer or a peer in *red* state as a next hop should not be in *red* state itself. The node which is not in *red* state is in *green* state. The *green* state is a domain stable state.

Performance evaluation

We used our generator of hierarchical topologies SHIIP *(Supélec Hierarchical Inter-domain Inferring Program)* [1], available at http://wwwsi.supelec.fr/~weisser/fr/shiip.html) to obtain network for experiences. We also choose the capacities in generated topologies to study networks which are not over-dimensioned. On the Internet, the largest domains with the largest capacities are at the top of the hierarchy. Domains with smaller capacities are at the bottom of the hierarchy. We suppose that the domains in the core are never perturbed. The first performance measure is obviously the number of perturbed nodes because it is also the optimization criterion for our algorithm. The second one is the number of perturbed paths and the third one is the amount of perturbed traffic. These three measures are essential for the comparison of performance results of our distributed algorithm with the results of BGP and the lowest theoretical bound. In Figure 1 we present the gain in perturbed node/path number obtained with our algorithm comparing to BGP routing [2].

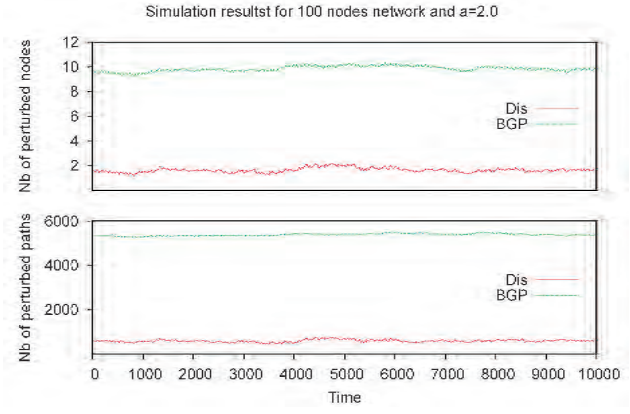

Figure 1: The number of perturbed nodes and paths averaged for a simulation run series, network of 100 nodes and heavy traffic.

References

[1] M.-A. Weisser, J. Tomasik, "Automatic Induction of Inter-Domain Hierarchy in Randomly Generated Network Topologies", 10th ACM/SIGSIM CNS'07, pp. 77-84, Norfolk, VA, USA, 26-28 March 2007.

[2] M.-A. Weisser, J. Tomasik, D. Barth, "Congestion avoiding mechanism based on inter-domain hierarchy", IFIP Networking 2008, LNCS 4982, pp. 470-481, Singapore, 5-9 May 2008.