



L'air de rien N° 12

Séminaires CSI de janvier

L'aléastriel du Laboratoire de Recherche et de Développement de l'EPITA¹

Numéro 12, janvier 2008

Édito

par *Alexandre Duret-Lutz (Enseignant-Chercheur)*

Bonne et heureuse année 2008 à tous !

Pour bien commencer cette nouvelle année, les étudiants de la promotion 2008 de l'option CSI présenteront leurs travaux à l'occasion de deux sémi-

naires, les 9 janvier² et 16 janvier³ 2008. Ces présentations sont ouvertes à tous et nous vous invitons à venir nombreux vous y faire une idée plus précise de nos thèmes de travail.

9 janvier 14h00–17h15 — salle Masters	
Olena	
14h00	Une introduction à SCOOP, un paradigme C++ orienté objet — Thomas Moulard
Diagrammes de décision	
14h30	Homolib — Samuel Charron
Vérification du locuteur	
15h00	SVM-MLLR et fusion pour la vérification du locuteur — Geoffroy Querol
Théorie des jeux	
15h45	Étude du fictitious play dans le cas d'un jeu à fonctions d'utilité identiques — Jean-Philippe Garcia Ballester
Vaucanson	
16h15	Booster la généricité de Vaucanson — Guillaume Lazzara
16h45	Transducteurs synchronisés — Guillaume Le-roi

16 janvier 14h00–17h15 — salle Masters	
Transformers	
14h00	revCPP : Un préprocesseur C++ réversible — Quentin Hocquet
14h30	Transformation de programmes en syntaxe concrète en C++ — Benoit Sigoure
15h00	Propagation automatique des attributs pour des grammaires attribuées modulaires — Nicolas Pierron
15h30	Désambiguïsation guidée par la sémantique : Comparaison de différentes méthodes — Renaud Durlin
Théorie des jeux	
16h15	La tablette de chocolat transfinie — Nicolas Neri
Vérification du locuteur	
16h45	Combinaison de noyaux à partir de systèmes SVM en vérification du locuteur — Charles Alban Deledalle

OLENA

Une introduction à SCOOP, un paradigme C++ orienté objet

par *Thomas Moulard* © 9 janvier, 14h00, salle Masters

Le C++ a réussi à supporter à la fois la programmation orienté objet classique et la programmation générique, cependant certains problèmes récurrents restent difficiles à résoudre. SCOOP est un paradigme orienté objet dont le but est de mélanger approche orienté objet classique et programma-

tion générique afin d'allier élégance, sécurité et rapidité. Le paradigme fournit des méthodes virtuelles, les arguments covariants, les types virtuels et les multi-méthodes typées statiquement sans avoir besoin d'étendre le langage. SCOOP fournit également des mécanismes d'écriture de morphers qui permettent d'exprimer en C++ des fonctions de type vers type. Cette présentation fait un tour d'horizon de SCOOP en présentant d'une part le paradigme en lui-même et d'autre part son utilisation en C++ au travers d'exemples.

¹L'air de rien, <http://publis.lrde.epita.fr/LrdeBulletin>.

²Séminaire CSI du 9 janvier 2008, <http://publis.lrde.epita.fr/Seminar-2008-01-09>.

³Séminaire CSI du 16 janvier 2008, <http://publis.lrde.epita.fr/Seminar-2008-01-16>.

DIAGRAMMES DE DÉCISION

Homolib

par *Samuel Charron* ☉ 9 janvier, 14h30, salle Masters

Les Diagrammes de Décision sont une famille de structures de données permettant de représenter avec peu de mémoire de grands ensembles de données. Ces structures peuvent être de taille fixe (un tuple) ou variable (une liste, un conteneur associatif, ...), la manipulation du DD ne se faisant pas de la même manière. Les Data Decision Diagrams et Set Decision Diagrams manipulent des données

de taille variable grâce à des opérations, les homomorphismes. Cependant la définition d'une opération correcte peut dérouter l'utilisateur, et passe souvent par de nombreuses erreurs, difficiles à identifier. Ce séminaire propose une bibliothèque d'algorithmes fournissant une vue plus abstraite que les homomorphismes "bruts" des données manipulées, en reprenant les algorithmes définis dans les modules "List" et "Map" d'Objective Caml. L'utilisateur peut se concentrer sur les parties spécifiques à son problème.

VÉRIFICATION DU LOCUTEUR

SVM-MLLR et fusion pour la vérification du locuteur

par *Geoffroy Querol* ☉ 9 janvier, 15h00, salle Masters

Afin d'améliorer la performance globale des systèmes de vérification du locuteur, il faut diversifier les approches. Le but de ce travail est d'étudier les performances d'un système SVM-MLLR. Cette méthode se base sur la construction, à partir du modèle du monde, d'une transformation linéaire des vecteurs moyennes (mean supervectors) maximisant la vraisemblance du modèle transformé par rapport aux données locuteur. On évaluera deux approches différentes : Dans la première, on utilisera directement le logarithme du rapport de vraisemblance (GMM-MLLR). Dans une deuxième expérimentation, on utilisera les SVMs pour évaluer les scores de décision. La dernière étape consiste à évaluer l'apport d'une méthode de compensation du canal (NAP : Nuisance Attribute Projection) sur les performances de ce système). Une fusion des scores avec d'autres systèmes GMM sera étudiée. Une fusion au niveau des noyaux sera quant à elle présentée par Charles-Alban. Tous les tests vont être menés sur les deux bases de données NIST-SRE 2005 et 2006 all trials.

Combinaison de noyaux à partir de systèmes SVM en vérification du locuteur

par *Charles Alban Deledalle* ☉ 16 janvier, 16h45, salle Masters

Les meilleurs systèmes de Vérification du Locuteur (VL) sont fondés sur la fusion des scores de décision de plusieurs approches. Les méthodes basées sur les Séparateurs à Vaste Marge (SVM) donnent des résultats très performants. En conséquence, l'apport de ces méthodes est très important pour la fusion. Dans notre approche, nous proposons une nouvelle méthode de fusion des systèmes de VL basés sur les méthodes SVM en construisant une nouvelle fonction noyau à partir d'une combinaison linéaire de plusieurs fonctions. Dans cette combinaison, les poids utilisés varient selon les locuteurs, ce qui diffère des approches par fusion de score qui elles utilisent des poids universels. L'idée est donc de tirer avantage des performances de chacun des noyaux, et cela pour chaque locuteur donné. Ces combinaisons sont effectuées sur plusieurs types de noyaux dont les noyaux GLDS, GMM supervecteurs linéaires et Gaussiens. Les expériences sont réalisées sur la base des corpus NIST-SRE 2005 et 2006.

THÉORIE DES JEUX

Étude du fictitious play dans le cas d'un jeu à fonctions d'utilité identiques

par *Jean-Philippe Garcia Ballester* ☉ 9 janvier, 15h45, salle Masters

Le fictitious play, en théorie des jeux, est une règle d'apprentissage dans laquelle chaque joueur suppose que ses adversaires jouent une stratégie fixe (potentiellement mixte, c'est-à-dire une distribution de probabilité sur un ensemble de stratégies). À chaque tour, chaque joueur joue ainsi le meilleur coup contre la stratégie de ses adversaires, déterminée de manière empirique à partir de leurs coups précédents. La convergence de telles stratégies n'est pas assurée, mais on sait que si il y a convergence, alors les stratégies jouées correspondront statistiquement à un équilibre de Nash. Il est donc très intéressant de connaître les critères de convergence. Nous nous intéresserons pour cette présentation au cas des jeux où les fonctions d'utilité (le gain d'un joueur en fonction des stratégies jouées) de chaque joueur sont identiques. Nous étudierons d'abord des résultats de

convergence dans ce cas particulier. Afin de réduire la complexité en temps, nous verrons une variante de cet algorithme, qui consiste à autoriser une erreur dans la meilleure réponse des joueurs. Nous présenterons enfin un exemple d'application du fictitious play pour résoudre un problème a priori non lié à la théorie des jeux : un problème d'optimisation, c'est-à-dire calculer le maximum des valeurs prises par une fonction.

La tablette de chocolat transfinie

par *Nicolas Neri* ☉ 16 janvier, 16h15, salle Masters

Nous nous attardons sur le jeu de la tablette de chocolat. On dispose d'une tablette de chocolat dont le carré inférieur gauche est empoisonné. Les joueurs jouent à tour de rôle. Un coup consiste à choisir un carré de chocolat et à le manger ainsi que tous les carrés qui sont à sa droite et au dessus de lui. Le joueur qui mange le carré empoisonné perd la partie. Dans cet exposé, nous nous intéresserons particulièrement au cas où les dimensions du jeu sont de classe cardinale infinie. On présentera également, pour une meilleure compréhension, les nombres ordinaux et leur ordre associé.

VAUCANSON

Booster la généricité de Vaucanson

par *Guillaume Lazzara* ☉ 9 janvier, 16h15, salle Masters

L'architecture du projet Vaucanson a été conçue initialement autour du design pattern Element. Ce dernier a l'énorme avantage de distinguer à la fois les concepts et les implémentations. C'est à dire que pour un type d'automate comme les automates booléens, on peut théoriquement avoir plusieurs implémentations qui se côtoient dans un même programme. Malgré toutes ces précautions, aujourd'hui, ajouter une nouvelle structure s'avère très délicat et remet en cause de nombreux points au sein du projet. C'est pour cette raison que durant ce séminaire nous tenterons de répondre à ces problèmes. Les problèmes de performances qu'a pu rencontrer le pro-

jet sont également une bonne motivation pour s'attaquer à ce sujet : il est aujourd'hui indispensable de proposer des nouvelles structures plus efficaces, notamment implémentées avec la bibliothèque Boost.

Transducteurs synchronisés

par *Guillaume Leroi* ☉ 9 janvier, 16h45, salle Masters

Lors de cette présentation, un algorithme de re-synchronisation sera décrit ainsi que son implémentation dans Vaucanson. De plus, des explications sont données sur l'ajout des transducteurs à délai borné, ainsi que sur les difficultés qui peuvent être rencontrées lors de l'extension de la hiérarchie de classes de Vaucanson.

TRANSFORMERS

revCPP un préprocesseur réversible

par *Quentin Hocquet* © 16 janvier, 14h00, salle Masters

Le but du projet Transformers est de créer un framework générique pour de la transformation source à source de code C++. Une transformation “source à source” consiste à retravailler le code et produire un fichier de code source modifié. Ce code peut être relu, réutilisé, modifié ... par des programmeurs et doit donc être lisible. De plus, il doit respecter le coding style d'origine. Ce processus de préservation du layout est appelé “High fidelity program transformation”. Contrairement à de nombreux langages, le C++ est un langage préprocessé pour obtenir le code source effectif. Dans le contexte de la transformation de programmes, il faut dé-préprocesser le code pour le rendre lisible au programmeur. Nous présentons le travail de recherche que nous avons mené pour implémenter un préprocesseur de C++ réversible et un postprocesseur, c'est-à-dire un outil permettant d'obtenir le code d'origine à partir du code préprocessé.

Propagation automatique d'attributs pour les grammaires attribuées modulaires

par *Nicolas Pierron* © 16 janvier, 15h00, salle Masters

Les grammaires attribuées sont plus adaptées pour décrire (des parties de) la sémantique d'un langage de programmation : accrochées sur les règles de production syntaxique, elles permettent d'exprimer des relations locales qui sont par la suite liées entre elles globalement par un évaluateur générique. Cependant elles ne passent pas à l'échelle quand on travaille avec des langages volumineux et complexes. Premièrement les attributs qui sont requis quasiment partout ont besoin d'être véhiculés par chaque règle de production. Deuxièmement, ces contraintes cassent la modularité car le fait d'étendre une grammaire nécessite la propagation des nouveaux attributs à travers le reste du langage. Ce papier montre comment résoudre ces problèmes en introduisant un système de propagation automatique des attributs qui complète l'ensemble

En bref

Une nouvelle publication (cf. publis.lrde.epita.fr) :

RICOU, O., BAILLARD, A., BERTIN, E., MAGNARD, F., MARMO, C., AND MELLIER, Y. Web services at TERAPIX. In *Proceedings of the XVII conference on Astronomical Data Analysis Software & Systems (ADASS)*. Cet article, une collaboration avec l'Insti-

des règles sémantiques. Nous avons défini formellement les contraintes de propagations de manière optimisée afin d'éviter l'ajout de règles sémantiques inutiles. Ainsi les grammaires attribuées sont devenues plus maintenables, modulaires et facile à utiliser.

Transformation de programmes en syntaxe concrète en C++

par *Benoît Sigoure* © 16 janvier, 14h30, salle Masters

La transformation de programmes dans des langages généralistes tels que le C++ est fastidieuse car elle nécessite de manipuler l'Abstract Syntax Tree (AST) du programme transformé en syntaxe abstraite dans le langage hôte (ici le C++). Le code à écrire est lourd et coûteux maintenir.

Nous présentons la mise en œuvre de nouvelles techniques de transformation de programmes en syntaxe concrète (c'est-à-dire utilisant directement le langage du programme transformé) dans un environnement C++ standard. Notre approche utilise l'analyseur syntaxique à l'exécution pour appliquer des règles de transformation dynamiques. Un compilateur de Tiger servira de support.

Désambiguïsation guidée par la sémantique

par *Renaud Durlin* © 16 janvier, 15h30, salle Masters

Modularité, extensibilité et expressivité, trois aspects fondamentaux pour un système de désambiguïsation. La désambiguïsation est l'étape survenant juste après l'analyse syntaxique qui consiste à analyser la sortie obtenue lors de l'utilisation d'un parseur LR généralisé. Cette étape étant sélectionne, parmi toute une forêt, l'unique arbre valide correspondant à l'entrée en prenant en compte les règles de sémantique contextuelles. Au travers d'une comparaison avec deux autres techniques reposant sur SDF (le formalisme ASF et le langage Stratego), le système de grammaires attribuées utilisé dans Transformers sera évalué par rapport à ces aspects fondamentaux pour en faire ressortir les avantages et inconvénients.

tut d'Astrophysique de Paris, présente les services de calcul mis à disposition de la communauté des astronomes dans le cadre de l'Observatoire Virtuel. L'architecture sous jacente utilisée pour la distribution des calculs est abordée.