

INSTITUTO DE COMPUTAÇÃO
UNIVERSIDADE ESTADUAL DE CAMPINAS

**Anais do 2º Workshop de Teses de Doutorado
em Andamento do IC-UNICAMP**

*Claudia M. Bauzer Medeiros, Amanda Meincke
Melo, Carla Geovana do Nascimento Macario,
Fernando Castor Filho, Neumar Costa Malheiros e
Vânia Paula de Almeida Neris (Eds.)*

Technical Report - IC-06-24 - Relatório Técnico

December - 2006 - Dezembro

The contents of this report are the sole responsibility of the authors.
O conteúdo do presente relatório é de única responsabilidade dos autores.

Apresentação

Este relatório técnico contém resumos de 32 trabalhos apresentados no II Workshop de Teses de Doutorado do Instituto de Computação da UNICAMP. O workshop, realizado de 29 de novembro a 1 de dezembro de 2006, permitiu que doutorandos do Instituto apresentassem os principais aspectos de suas pesquisas. Cada capítulo corresponde a uma apresentação, sendo o texto limitado a 3 páginas.

A participação foi voluntária e o perfil acadêmico dos participantes foi variado, cobrindo desde alunos recém-admitidos no programa até aqueles que já tinham defesa marcada em dezembro de 2006.

A publicação dos resumos sob forma de um relatório técnico tem por objetivo uma maior divulgação de trabalhos de doutorado em andamento no IC. Além disso, é um registro sucinto do estado de várias dessas pesquisas.

Como coordenadora do Workshop, destaco o trabalho dos cinco alunos de doutorado que foram os efetivos organizadores do evento e que são co-editores deste relatório - Amanda, Carla, Fernando, Neumar e Vânia. Ressalto, igualmente, o trabalho de todos os autores de capítulos, que se dispuseram a apresentar trabalhos em andamento e seus orientadores que incentivaram tal participação.

Finalmente, agradeço à coordenação de pós-graduação do IC pelo apoio.

Claudia Bauzer Medeiros
Coordenadora do 2º WTD
Professora - Instituto de Computação - UNICAMP

Sumário

1 Test Generation by Model Checking Timed Extended Finite State Machines	
Adilson Luiz Bonifácio, Arnaldo Vieira Moura, Adenilso da Silva Simão, José Carlos Maldonado	5
2 Transactional Memory	
Alexandro Baldassin and Paulo Centoducatte	8
3 Detecção e Localização de Displasias em Imagens de Ressonância Magnética do Cérebro Humano	
Felipe P.G. Bergo e Alexandre X. Falcão	11
4 Sumarização Automática de Prontuários Eletrônicos e Geração de Textos Médicos	
Cleo Zanella Billa e Jacques Wainer	14
5 Sistemas Reativos Distribuídos na Tecnologia Xchart	
Carlos Neves Júnior e Hans Kurt Edmund Liesenberg	17
6 A framework based on semantic Web services and AI planning for the management of bioinformatics scientific workflows	
Luciano Antonio Digiampietri, João C. Setubal, and Claudia B. Medeiros	20
7 Uma abordagem em camadas para especificação de políticas de segurança de sistemas operacionais para ambientes heterogêneos	
Diogo Ditzel Kropiwiec, Paulo Lício de Geus	23
8 Dynamic Traffic Grooming in IP over WDM Networks with Quality of Service	
André Costa Drummond and Nelson L. S. da Fonseca	26
9 O método da geração de colunas aplicado ao problema do separador de vértices em grafos	
Edna Ayako Hoshino e Cid C. de Souza	29
10 Contract Negotiation on Agricultural Supply Chains	
Evandro Bacarin, Edmundo R.M. Madeira, and Claudia Bauzer Medeiros	32
11 Detecção de anomalias em <i>logs</i> de sistemas de apoio a processos de negócios	
Fábio de Lima Bezerra e Jacques Wainer	35

12 Workflow-based Sensor Data Management	
Gilberto Zonta Pastorello Jr. and Claudia Bauzer Medeiros	38
13 Composicionalidade de Escalonamento de Processos em Sistemas de Workflow	
Gregório Baggio Tramontina e Jacques Wainer	41
14 Software Architectures for Distributed Main-Memory Databases	
Gustavo Maciel Dias Vieira, Anderson Supriano, and Luiz Eduardo Buzato	44
15 WiMAX - Acesso à Internet com Qualidade de Serviço	
Juliana Freitag Borin e Nelson L. S. da Fonseca	47
16 Análise/Otimização de Potência de SoCs Heterogêneos Ponderando	
Felipe Klein e Rodolfo Azevedo	50
17 Algoritmo para Problema de Conexidade em Grafos	
Luís Augusto Angelotti Meira e Flávio Keidy Miyazawa	53
18 Design Inclusivo de Sistemas de Informação na Web	
Amanda Meincke Melo, M. Cecilia C. Baranauskas	56
19 Uma Arquitetura para Redes Mesh Sem Fio Auto-Gerenciáveis	
Neumar Malheiros e Edmundo Madeira	59
20 Replicação de Checkpoints em Ambientes Distribuídos	
Raphael Marcos Menderico, Islene Calciolari Garcia	62
21 Criptografia Baseada em Emparelhamentos em Redes de Sensores Sem Fio	
Leonardo B. Oliveira, Ricardo Dahab	65
22 Uma Metodologia de Desenvolvimento Rigoroso Centrado na Arquitetura para Sistemas Baseados em Componentes	
Patrick Henrique da Silva Brito e Cecília Mary Fischer Rubira	68
23 Algoritmos para Problemas de Particionamento em Grafos	
Evandro Cesar Bracht, Flávio K. Miyazawa e Orlando Lee	71
24 The 2D-VLIW Architecture: Execution Model and Code Generation	
Ricardo Santos, Rodolfo Azevedo, and Guido Araujo	74

25 Uma Ponte entre as Abordagens Síncrona e Quase-síncrona para Checkpointing	
Tiemi Christine Sakata e Islene Calciolari Garcia	77
26 Coloração de Arestas: o Problema da Classificação	
Sheila de Almeida e Célia P. de Mello	80
27 Geração de Dados para o Teste de Conformidade baseado em MEF Estendida	
Thaise Yano, Eliane Martins e Fabiano Luís de Sousa	83
28 Relax-and-Cut algorithms to 0-1 Combinatorial Optimization problems	
Victor Fernandes Cavalcante and Cid Carvalho de Souza	86
29 Algoritmos para Problemas de Empacotamento	
Eduardo C. Xavier e Flávio K. Miyazawa	89
30 Tratamento de Exceções no Desenvolvimento de Sistemas Tolerantes a Falhas Baseados em Componentes	
Fernando Castor Filho e Cecília Mary F. Rubira	91
31 Sumarização automática para vídeos de futebol televisivos	
Nielsen Cassiano Simões, Neucimar J. Leite, and Beatriz Marcotegui	94
32 Reconhecimento de Parasitos Intestinais do Homem em Imagens de Microscopia Ótica	
João Paulo Papa e Alexandre Xavier Falcão	96

1. Test Generation by Model Checking Timed Extended Finite State Machines

Authors: Adilson Luiz Bonifácio, Arnaldo Vieira Moura, Adenilso da Silva Simão, José Carlos Maldonado

Model-based testing comprises the automatic generation of efficient test cases using mathematical models of system requirements, and is based on formally specified system functionalities. However, determining whether or not an implementation is equivalent to its specification is infeasible for most practical applications. Thus, this equivalence is relaxed into a conformance relation, named conformance testing [11].

The use of conformance testing for the problem of generating test cases based on Finite State Machines (FSMs) has been intensively investigated [5, 9]. Nonetheless, there are many situations in which the modeling of the system as a FSM is cumbersome, due to the state explosion problem, or even impossible, due to the fact that there are some relevant aspects that can not be properly expressed, e.g., time. Thus, some extensions to the FSM model have been proposed in order to overcome these problems [1, 4, 12]. Some of them incorporate notions like context variables and input/output parameters. Others incorporate notions of time, such as Timed Transition Systems (TTSs), allowing the model to capture the evolution of time.

An Extended FSM (EFSM) can be thought of as a folded FSM. Given an EFSM [10], and assuming that the domains are finite, it is possible to unfold it into a pure FSM by expanding the values of its parameters and variables. The resulting FSM can be used with FSM-based methods for test derivation with complete fault coverage, which means all fault possibilities would be exhausted. However, in practical situations, this approach is not feasible, mainly due to the state explosion effect [10].

In many system categories, time plays an important role in determining the acceptability of system behavior. Not only the input/output relationship is relevant, but also the period of time when those events occur may be important. In such cases, it is imperative to be able to represent the time constraints of the system, and to test whether a given implementation conforms to these constraints. There are some formalisms that allow the representation of various time related concepts, such as TTS [5] and Timed Automata [1, 2, 6]. Nonetheless, there are few, if any, methods that allow a satisfactory derivation of adequate test cases from those models.

In a broad context, our research aims at model-based methods for testing systems with time constraints. In particular, we are addressing the problem of the construction of an adequate formalism for modeling systems and the automatic generation of test cases, as well as the determination of the expected outputs. These tasks are closely related, and should be considered together.

In this work, we define Timed EFSMs (TEFSMs) by including the notion of explicit and implicit time advancement, by using Extended TTSs (ETTSS), in the

EFSM formalism. We can adapt and employ some results that are well-established in model-based testing from (E)FSMs to the context of systems that require time constraints. In particular, we address the problem of configuration confirmation for TEFSMs in the same vein as done in [10]. In that work, it is shown how the problem of configuration confirmation for EFSMs can be reduced to the problem of finding a path in an EFSM product. By defining a property that states when no such a path exists, model-checking techniques can be used to generate a confirming sequence.

Configuration confirming sequences (CCSs) can be derived from the product of two TEFSMs. A CCS is a sequence of parameterized timed inputs that can be applied to the implementation in order to increase the confidence that the correct configuration has been reached in the implementation, by showing that no undesirable configuration has been reached. Given a configuration and a set of suspicious configurations, a confirming sequence is a sequence of (parameterized) inputs that allows us to distinguish the given configuration from configurations in the set of suspicious configurations by comparing outputs and, possibly, observing the time indicated in each of the outputs. However, unlike the FSMs, where a finite set of undesirable configurations can be postulated, with EFSMs and TEFSMs it is not possible, or desirable, to determine all undesirable configurations. To overcome this problem, a set of suspicious configurations is considered [10]. A set of suspicious configurations is derived from the specification to model implementations which can potentially have faults, reflecting the test designer’s assumptions about the implementation faults. The suspicious configurations are extracted from the specification using a set of test hypotheses based on the fault model (e.g., [7]) and relying on the test designer expertise. Finding a confirming sequence can also be seen as an extension of the state identification problem [8].

The ability to derive test cases from formal models opens the possibility that we can construct more rigorous and dependable systems, by providing a sound basis for the validation of the systems’ behaviors. There is a direct relationship between the kinds of systems that a given model can deal with and the availability of methods for deriving test cases. FSM and EFSM are well-established models that have been intensively investigated. One important feature they both lack is the ability to deal with time. In this paper we defined the TEFSM as a model that extends the EFSM model with the notion of time, as in the ETTS model. From that, we discussed an extended method for deriving configuration confirming sequence for TEFSMs, thereby automating the generation of test cases for these models [3].

Although we can argue that both our model and our generation method can be used, we do not have answers for pragmatic questions, such as (i) how difficult is it to describe a system using TEFSMs and (ii) how large are the models we can handle. To answer these question, it is necessary to deepen the investigations and implement the adequate software tools. We are currently working in this direction.

Other aspects that can be investigated include how to allow time constraint to be defined over outputs. We note that our definition does not deal with constraints that may reflect output response that is not instantaneous. The input and output

occur in the same time instant. We are considering how this extension might impact the test case generation methods.

Referências

- [1] R. Alur. Timed automata. In *CAV'99*, number 1633 in LNCS, 1999.
- [2] Rajeev Alur and David L. Dill. A theory of timed automata. *Theoretical Computer Science*, 126(2):183–235, 1994.
- [3] Adilson Luiz Bonifácio, Arnaldo Vieira Moura, Adenilso da Silva Simão, and José Carlos Maldonado. Conformance Testing by Model Checking Timed Extended Finite State Machines. In *Brazilian Symposium on Formal Methods (SBMF)*, pages 43–58, Natal-RN, Brazil, September 2006.
- [4] S. V. Campos, M. Minea, W. Marrero, E. M. Clarke, and H. Hiraishi. Computing quantitative characteristics of finite-state real-time systems. In *Proc. 15th IEEE Real-Time Systems Symp.*, pages 266–270. IEEE Comp. Soc. Press, Dec 1994. San Juan, Porto Rico.
- [5] Rachel Cardell-Oliver. Conformance tests for real-time systems with timed automata specifications. *Formal Aspects of Computing*, 12(5):350–371, 2000.
- [6] Martin Dickhöfer and Thomas Wilke. Timed alternating tree automata: the automata-theoretic solution to the tctl model checking problem. In *26th ICALP*, volume 1644 of LNCS, pages 281–290, 1999.
- [7] Abdeslam En-Nouaary, Rachida Dssouli, and Ferhat Khendek. Timed wp-method: Testing real-time systems. *IEEE Trans. Softw. Eng.*, 28(11):1023–1038, 2002.
- [8] M. Krichen. State identification. In M. Broy, B. Jonsson, J.-P. Katoen, M. Leucker, and A. Pretschner, editors, *Model-Based Testing of Reactive Systems: Advanced Lectures*, volume 3472 of *Lecture Notes in Computer Science*, pages 87–111. Springer-Verlag, 2005.
- [9] M. Krichen and S. Tripakis. Black-box conformance testing for real-time systems. In *Model Checking Software: 11th International SPIN Workshop*, number 2989 in *Lecture Notes in Computer Science*, pages 109–126, Barcelona, Spain, April 2004.
- [10] Alexandre Petrenko, Sergiy Boroday, and Roland Groz. Confirming configurations in efsm testing. *IEEE Trans. Softw. Eng.*, 30(1):29–42, 2004.
- [11] Jan Tretmans. Test generation with inputs, outputs, and quiescence. In Tiziana Margaria and Bernhard Steffen, editors, *Tools and Algorithms for Construction and Analysis of Systems (TACAS), Passau, Germany, March 27-29*, volume 1055 of *Lecture Notes in Computer Science*, pages 127–146. Springer, 1996.
- [12] Farn Wang. Formal verification of timed systems: A survey and perspective. *Proceedings of the IEEE*, 92(8):1283–1307, 2004.

2. Transactional Memory

Authors: Alexandro Baldassin and Paulo Centoducatte

Concurrent programming is inherently complex and difficult. Each execution of the same program with the same input set might produce a myriad of different traces, making it extremely hard to test and debug. Traditional synchronization approaches based on mutual exclusion locks and condition variables do not perform well on symmetric multiprocessor architectures (SMP), where two or more processors share the same global physical address space. The problem with the aforementioned approaches is mainly due to the serial bottleneck imposed by the locking mechanism to protect critical sections, which prevents processes to proceed concurrently even when they could. Moreover, locking techniques might also lead to common problems such as priority inversion, convoying and deadlock.

In order to ease the programming of concurrent systems and, at the same time, boost their performance on shared memory multiprocessors, researchers have proposed *transactional memory*. The transaction concept is not new at all, being a fundamental concept in database systems for a long time [7]. A transaction is a group of instructions that has the *atomicity* and *serializability* properties: its changes are either made available to the rest of the system all at once (*committed*) or are discarded without being noticed (*aborted*); transactions are executed serially so that instructions from one transaction do not appear to be interleaved with instructions from another one. To achieve higher performance, a transaction executes optimistically without taking locks; every memory reference is recorded into a transaction log that must be validated before the transaction commits. Through a lock-free synchronization approach, transactions provide higher and more intuitive language constructs while at the same time augmenting the level of concurrency on the system.

In this work, we briefly present references and developments in transactional memory. For space reasons we do not discuss in detail each single paper in the literature, but we do provide pointers to the most important ones. The number of publications on the subject has increased substantially in the last and current years, and it seems unlikely to slow down so soon.

In 1986, Tom Knight proposed a framework in which a serially-written Lisp program could be executed in parallel [15]. His compiler transformed a program into groups of primitive instructions, so named *transaction blocks*. Each transaction block was guaranteed to be independent of another (except for main memory contents) and had exactly one side effect, occurring at the end of the block. The runtime system was responsible to schedule the execution of the blocks in a consistent manner, possibly on several different processors. If consistency failed for a block, it had to be *aborted* and *rolledback*. In order to maintain and enforce dependencies among transactions, Knight proposed the use of two fully associative caches, thus using a *hardware transactional memory* approach. In 1988, Tinker et al. [25] presented a technique to parallelize execution of an Scheme program, but Tinker's approach was based on software. Currently, the Haskell functional programming language is

receiving much attention by transactional memory researchers [10] [6]. There is even a software transactional memory implementation available as part of the Glasgow Haskell compiler library.

Herlihy and Moss published their seminal paper on transactional memory in 1993 [14]. They proposed a mechanism to implement transactions on hardware, by extending the multiprocessor cache-coherence protocol and instruction set architecture. Their lock-free approach to synchronization performed better than its lock-based counterpart as showed by their results, while at the same time avoiding deadlock, priority inversion and convoying. A similar idea was concurrently explored by Stone et al. [24], with the differences varying in the implementation aspect. Two years later, based on the work of Herlihy and Moss, Shavit and Touitou proposed a software method to implement transactional memory [23]. Their paper is the first well known one to describe an implementation of a *software transactional memory*. Although their method was far more flexible (software), it considered only static transactions and required a strong atomic primitive which is still not supported by today's processor standards.

After Shavit and Tiuitou's paper there was not much development in transactional memory until circa 2003, when the topic came under the spotlight again. At one side, hardware-based approaches attempted to overcome limitations in the work of Herlihy and Moss, such as transaction size, and, at the same time, proposed new buffering and conflict resolution mechanisms. It is worth mentioning the works from Stanford University [8] [17] [4], MIT [2], Intel Corporation [20] and University of Wisconsin-Madison [18] [19]. At the other side, software-based approaches introduced dynamic transactions and provided new language constructs to describe concurrent programs. The main works include those from Stanford University [3], Sun Microsystems and Brow University [13] [12], Microsoft and University of Cambridge [9] [11], University of Rochester [22] and Intel Corporation [21] [1]. Hardware and software approaches are discussed together in [16] and [5].

Referências

- [1] Ali-Reza Adl-Tabatabai, Brian T. Lewis, V. Menon et al. Compiler and runtime support for efficient software transactional memory. In *PLDI'06*, pages 26–37.
- [2] C. S. Ananian, K. Asanovic, B. C. Kuszmaul, C. E. Leiserson, and Sean Lie. Unbounded transactional memory. *IEEE Micro*, 26(1):59–69, January/February 2006.
- [3] B. D. Carlstrom, A. McDonald, H. Chafi, J. Chung, Chi Cao Minh, Christos Kozyrakis, and K. Olukotun. The Atomos transactional programming language. In *PLDI'06*, pages 1–13.
- [4] Weihaw Chuang, Satish Narayanasamy, Ganesh Venkatesh et al. Unbounded page-based transactional memory. In *ASPLOS-XII*, pages 347–358, October 2006.
- [5] Peter Damron, Alexandra Fedorova, and Yossi Lev. Hybrid transactional memory. In *ASPLOS-XII*, pages 336–346, October 2006.

- [6] A. Discolo, Tim Harris, S. Marlow, S. Peyton-Jones, and S. Singh. Lock free data structures using STM in Haskell. *Lectures Notes in Computer Science*, Volume 3945:65–80, 2006.
- [7] Jim Gray. The transaction concept: Virtues and limitations. In *VLDB'81*, pages 144–154.
- [8] Lance Hammond, Vicky Wong, Mike Chen, Brian D. Carlstrom, John D. Davis et al. Transactional memory coherence and consistency. In *ISCA '04*, pages 102–113.
- [9] Tim Harris and Keir Fraser. Language support for lightweight transactions. In *OOPSLA '03*, pages 388–402.
- [10] Tim Harris, Simon Marlow, Simon Peyton-Jones, and Maurice Herlihy. Composable memory transactions. In *PPoPP'05*, pages 48–60.
- [11] Tim Harris, Mark Plesko, Avraham Shinnar, and David Tarditi. Optimizing memory transactions. In *PLDI'06*, pages 14–25.
- [12] Maurice Herlihy, Victor Luchangco, and Mark Moir. A flexible framework for implementing software transactional memory. In *OOPSLA '06*, pages 253–262.
- [13] Maurice Herlihy, Victor Luchangco, Mark Moir, and William N. Scherer. Software transactional memory for dynamic-sized data structures. In *PODC'03*, pages 92–101.
- [14] Maurice Herlihy and J. Eliot B. Moss. Transactional memory: Architectural support for lock-free data structures. In *ISCA '93*, pages 289–300.
- [15] Tom Knight. An architecture for mostly functional languages. In *LFP'86*, pages 105–112.
- [16] Sanjeev Kumar, Michael Chu, Christopher J. Hughes, Partha Kundu, and Anthony Nguyen. Hybrid transactional memory. In *PPoPP'06*, pages 209–220.
- [17] A. McDonald, J. Chung, B. D. Carlstrom, Chi Cao Minh, H. Chafi et al. Architectural semantics for practical transactional memory. In *ISCA '06*, pages 53–65.
- [18] K. E. Moore, J. Bobba, M. J. Moravan, M. D. Hill, and D. A. Wood. LogTM: Log-based transactional memory. In *HPCA'06*, pages 254–265.
- [19] M. J. Moravan, J. Bobba, K. E. Moore, Luke Yen, M. D. Hill, Ben Liblit et al. Supporting nested transactional memory in LogTM. In *ASPLOS-XII*, pages 359–370, October 2006.
- [20] Ravi Rajwar, Maurice Herlihy, and K. Lai. Virtualizing transactional memory. In *ISCA '05*, pages 494–505.
- [21] Bratin Saha, A. Adl-Tabatabai et al. McRT-STM: A high performance software transactional memory system for a multi-core runtime. In *PPoPP'06*, pages 187–197.
- [22] William N. Scherer and Michael L. Scott. Advanced contention management for dynamic software transactional memory. In *PODC'05*, pages 240–248.
- [23] Nir Shavit and Dan Touitou. Software transactional memory. In *PODC'95*, pages 204–213.
- [24] J. M. Stone, H. S. Stone, P. Heidelberger, and J. Turek. Multiple reservations and the Oklahoma update. *IEEE Parallel & Distributed Technology: Systems & Application*, 1(4):58–71, November 1993.
- [25] Pete Tinker and M. Katz. Parallel execution of sequential Scheme with ParaTran. In *LFP'88*, pages 28–39.

3. Detecção e Localização de Displasias em Imagens de Ressonância Magnética do Cérebro Humano

Autores: Felipe P.G. Bergo e Alexandre X. Falcão

Displasias corticais focais (FCDs) [10] são uma das principais causas de epilepsias resistentes ao tratamento medicamentoso, e seu diagnóstico não invasivo é um procedimento complexo, demorado e subjetivo [2, 8, 11]. Propomos um método automático de detecção de displasias a partir de imagens de ressonância magnética (RM). O método baseia-se na análise automática de uma reformatação curvilínea do cérebro, computada de forma também automática. O resultado final do método é uma visualização interativa do cérebro com possíveis localizações de displasias classificadas por probabilidade, permitindo uma inspeção rápida onde há indícios de lesões.

FCDs são lesões congênitas do córtex cerebral, caracterizadas pela distribuição anormal de neurônios nas camadas corticais [10]. Epilepsia é uma condição neurológica crônica caracterizada por crises espontâneas recorrentes. A presença de FCDs é a segunda causa mais comum de epilepsias resistentes ao tratamento medicamentoso [11].

Trabalhos Anteriores. As FCDs foram descritas por Taylor [10] na década de 1970. A partir de 1990 a técnica de RM passou a oferecer exames volumétricos com resolução adequada, e os indícios de FCDs puderam ser detectados visualmente de forma não invasiva. Este ainda é o método mais comum na prática clínica. Lesões sutis requerem tratamento especial das imagens [2], realizado de forma interativa e não repetível. Diversos trabalhos recentes propõem métodos para detecção automática de FCDs, porém com falhas metodológicas ou premissas inadequadas [1, 6, 9], com taxas de acerto entre 71% e 85%.

Proposta. Nossa proposta é um método automático para classificar possíveis localizações de displasias a partir de imagens tridimensionais de RM. Como resultado final, o método desenvolvido permitirá a visualização tridimensional interativa de reformatações curvilíneas [2] com indicações gráficas das localizações de possíveis displasias, permitindo que o médico inspecione apenas as regiões suspeitas em vez de centenas de imagens.

Nosso método é composto por 5 operações: (1) segmentação do cérebro, (2) correção de alinhamento, (3) mapeamento de cada superfície curvilínea para um sistema de coordenadas topologicamente bem definido, (4) análise de textura e classificação de lesões e (5) visualização interativa do resultado.

A segmentação do cérebro é realizada através da técnica de Tree Pruning [3, 7], que é tolerante a ruídos e independente de modelos (templates), sendo aplicável a

pacientes de qualquer idade. A correção de alinhamento é realizada com uma otimização local de um plano de divisão entre os hemisférios. O mapeamento cada superfície utiliza uma transformada de distância Euclideana baseada na Transformada Imagem-Floresta [5] e em uma transformação simples de coordenadas esféricas para retangulares.

Displasias aparecem como borramentos assimétricos nas imagens planarizadas. A análise de textura usa medidas de textura e medidas no domínio da frequência para alimentar um classificador baseado em grafos. A visualização 3D das lesões detectadas pode ser facilmente realizada com o hardware atualmente disponível [4].

O método proposto não requer a segmentação de WM e GM, independe de templates e automatiza o procedimento visual de detecção de displasias por especialistas [2,8], que se baseia na identificação de assimetrias de textura entre os hemisférios do cérebro em superfícies que acompanham sua curvatura.

O método será validado em um banco de imagens disponibilizado pela FCM-Unicamp, que inclui tanto pacientes com FCDs confirmadas como controles sadios. Nosso objetivo é detectar 100% das FCDs, ainda que sejam detectados alguns falsos positivos, já que o diagnóstico final será dado por especialistas.

Referências

- [1] SB Antel et al. Automated detection of focal cortical dysplasia lesions using computational models of their MRI characteristics and texture analysis. *NeuroImage*, 19(4):1748–1759, Aug 2003.
- [2] AC Bastos et al. Diagnosis of subtle focal dysplastic lesions: Curvilinear reformatting from three-dimensional magnetic resonance imaging. *Annals of Neurology*, 46(1):88–94, 1999.
- [3] FPG Bergo and AX Falcão. Fast and automatic curvilinear reformatting of MR images of the brain for diagnosis of dysplastic lesions. In *Proc. of 3rd Intl. Symp. on Biomedical Imaging*, pages 486–489. IEEE, Apr 2006.
- [4] F de Goes, FPG Bergo, AX Falcão, S Goldenstein, and L Velho. Adapted dynamic meshes for deformable surfaces. In *XIX Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI)*, pages 213–220. IEEE, Oct 2006.
- [5] AX Falcão, J Stolfi, and RA Lotufo. The image foresting transform: Theory, algorithms, and applications. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26(1):19–29, 2004.
- [6] J Kassubek, HJ Huppertz, J Spreer, and A Schulze-Bonhage. Detection and localisation of focal cortical dysplasia by voxel-based 3-D MRI analysis. *Epilepsia*, 43(6):596–602, Jun 2002.

- [7] PAV Miranda, FPG Bergo, LM Rocha, and AX Falcão. Tree-pruning: A new algorithm and its comparative analysis with the watershed transform for automatic image segmentation. In *XIX Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI)*, pages 37–44. IEEE, Oct 2006.
- [8] MA Montenegro, LM Li, MM Guerreiro, CAM Guerreiro, and F Cendes. Focal cortical dysplasia: Improving diagnosis and localization with magnetic resonance imaging multiplanar and curvilinear reconstruction. *J Neuroimaging*, 12(3):224–230, Jul 2002.
- [9] S Srivastava et al. Feature-based statistical analysis of structural MR data for automatic detection of focal cortical dysplastic lesions. *NeuroImage*, 27(2):253–266, Aug 2005.
- [10] DC Taylor, MA Falconer, CJ Bruton, and JA Corsellis. Focal dysplasia of the cerebral cortex in epilepsy. *J Neurol Neurosurg Psychiatry*, 34(4):369–387, Aug 1971.
- [11] H Urbach, J Hattingen, J von Oertzen, C Luyken, H Clusmann, T Kral, M Kurthen, J Schramm, I Blümcke, and HH Schild. MR imaging in the pre-surgical workup of patients with drug-resistant epilepsy. *Am J Neuroradiol*, 25(6):919–926, Jun 2004.

4. Sumarização Automática de Prontuários Eletrônicos e Geração de Textos Médicos

Autores: Cleo Zanella Billa e Jacques Wainer

Sumarização de Prontuários Eletrônicos

A primeira parte desse trabalho trata de sumarização automática de prontuários eletrônicos, que visa diminuir o tempo do profissional da área de saúde na extração de informações de um grande conjunto de dados.

Em [1,3] são apresentadas diversas propostas de sumarização de textos médicos. A sumarização na área médica possui diversos desafios, um deles é a variedade de fontes de informação, por isso os vários trabalhos que existem na área acabam se destinando a um problema específico. Com relação a sumarização de prontuários eletrônicos, [2] apresenta a proposta de um sumário que faz inferência de diagnósticos a partir dos exames realizados e gera gráficos com a probabilidade de cada diagnóstico. Já [4] utiliza o formato de árvores para gerar uma visão que sumariza as informações clínicas identificadas através de processamento de linguagem natural.

De acordo com [1], o sumarizador aqui proposto é: de múltiplos documentos, porque analisa diversas consultas e gera apenas um sumário; indicativo, ou seja, não tem a pretensão de substituir o registro eletrônico do paciente, mas sim, ser uma alternativa para auxiliar o médico; e de apenas um idioma, já que trabalha apenas com textos em português, tanto para análise das informações quanto para a geração do sumário. Além disso, junto com o sumário em formato texto, é gerado um gráfico que apresenta as informações de forma mais rápida e direta.

Segundo [3] o processo de sumarização é dividido em três passos: análise da fonte, seleção do conteúdo e a geração do sumário em si.

Análise Da Fonte

Para a realização desse projeto foi utilizado uma base de dados cedida pelo ambulatório Centro-Alfa da UNIFESP. Ao examinar a base de dados, foi constatado que o registro eletrônico do paciente contém os exames físicos, exames laboratoriais e medicamentos prescritos na forma estruturada e portanto podem ser facilmente recuperados. Mas foi visto também que os sintomas do paciente não estão disponíveis de forma estruturada, eles só estão descritos no histórico clínico do paciente, que é um campo escrito em linguagem natural.

Seleção do Conteúdo

Para cada tipo de dado é dado um tratamento diferente. Os diagnósticos só fazem parte do sumário na data que são confirmados pelo médico. Todos os exames físicos ou laboratoriais entram no sumário em sua respectiva data. Já os medicamentos só aparecem no sumário na data em que foram prescritos ou que foram retirados do tratamento.

Para recuperar os sintomas do paciente é feita uma análise no histórico clínico do paciente. Como o histórico é um campo escrito em linguagem natural, é preciso mais do que simplesmente procurar por uma palavra dentro do texto, porque no histórico o médico registra se o paciente apresentou o sintoma ou não. Por exemplo, é comum frases do tipo: “O paciente nega tonturas, mas se queixa de dores de cabeça.”. Para extrair os sintomas do histórico, é feita uma análise nas sentenças para descobrir se o paciente apresenta ou não o sintoma. O sistema divide os sintomas em dois grupos: os sintomas apresentados e os não apresentados pelo paciente. Os dois grupos são inseridos no sumário a cada consulta.

Geração de Texto

A saída do sistema é um texto corrido que contém os sintomas apresentados e não apresentados pelo paciente, os exames realizados e seus resultados, os diagnósticos dados e os medicamentos prescritos. Além da criação do sumário em formato texto, é criado um gráfico que contém uma linha de tempo, onde essas mesmas informações são apresentadas. O gráfico criado não precisa do mesmo tipo de processamento dado ao texto, já que as informações são apresentadas de forma direta.

Resultados

A avaliação dos sumário será feita por profissionais da área de saúde da UNIFESP que irão receber o sumário gerado pelo sistema e o histórico completo do paciente. A partir desses dois dados eles irão julgar a completude do sumário. Essa avaliação está sendo realizada no presente momento, portanto os resultados ainda não estão disponíveis.

Geração de Textos Médicos a partir de Raciocínio de Projeto

A segunda parte desse projeto tem como objetivo a geração de texto a partir dos dados presentes no registro eletrônico do paciente.

Todo profissional da área de saúde, após uma consulta, deve elaborar um texto informando quais foram os exames realizados, os exames laboratoriais pedidos, os medicamentos prescritos, etc. Esse texto deve ser assinado e deixado na instituição como registro do que foi feito.

Para gerar esse texto, a proposta é que se compreenda como o profissional da área de saúde realiza a consulta. Portanto, para gerar um texto coerente com a consulta é necessário que se compreenda o porquê do profissional da área de saúde ter tomado uma determinada decisão. Por isso, propõem-se a utilização de *design rationale* ou raciocínio de projeto.

Design rationale possui várias definições, de acordo com o contexto que é utilizado. [5] apresenta algumas dessas definições, dentre elas uma diz que *design rationale* são as razões lógicas dadas para justificar o artefato projetado. No nosso caso, *design rationale* é a noção de que uma consulta vai além de meramente exames, diagnósticos e medicamentos, é também as razões e o processo de raciocínio do profissional da área de saúde.

Existem diversas abordagens para representar *design rationale*. [7] apresenta o

gIBIS que é uma ferramenta gráfica para a representação de *design rationale* baseada no método IBIS (*Issue Based Information Systems*). Já [6] sugere um método denominado QOC (*Questions, Options and Criteria*) que permite a representação das escolhas tomadas durante o projeto e suas justificativas.

Para esse trabalho, o primeiro passo é a criação de uma interface onde o profissional da área de saúde possa explicitar qual foi o raciocínio para realizar uma determinada ação, que pode ser um diagnóstico, a prescrição de um medicamento, etc. A idéia é que a interface permita que ele possa relacionar todas as informações por ele inserida, criando assim uma representação de *design rationale*.

O segundo passo é a geração do texto da consulta a partir da descrição representada no *design rationale*. O que permitiria um texto mais fiel e mais explicativo sobre acontecimentos da consulta.

Referências

- [1] Stergos Afantenos, Vangelis Karkaletsis, and Panagiotis Stamatopoulos. Summarization from medical documents: a survey. *Artificial Intelligence in Medicine*, 33:157–177, Fevereiro 2005.
- [2] S. M. Downs, M. G. Walker, and R. L. Blum. Automated summarization of on-line medical records. *Medical Informatics*, pages 800–804, 1986.
- [3] D. B. Johnson, Q. Zou, J. D. Dionisio, V. Zhenyu Liu, and W. W. Chu. Modeling medical content for automated summarization. *Annals of the New York Academy of Sciences*, pages 247–258, 2002.
- [4] H. Liu and C. Friedman. Cliniviewer: A tool for viewing electronic medical records based on natural language processing and xml. *Medical Informatics*, pages 639–643, 2004.
- [5] T. P. Moran and J. M. Carroll. Overview of Design Rationale. *Design Rationale: Concepts, Techniques, and Use*. Lawrence Erlbaum Associates, Mahwah, New Jersey, 1996.
- [6] A. MacLean and R. M. Young and T. P. Moran Design rationale: the argument behind the artifact. *CHI '89: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 247–252, 1989.
- [7] J. Conklin and M. L. Begeman gIBIS: a hypertext tool for team design deliberation. *HYPERTEXT '87: Proceeding of the ACM conference on Hypertext*, pages 247–251, 1987.

5. Sistemas Reativos Distribuídos na Tecnologia Xchart

Autores: Carlos Neves Júnior e Hans Kurt Edmund Liesenberg

O estudo de sistemas reativos é um tema que vem sendo discutido ao longo dos anos e a literatura de engenharia de software e sistemas é unânime ao reconhecer que existe uma grande dificuldade para especificação de sistemas reativos com grande complexidade e tamanhos. Esta dificuldade motivou alguns trabalhos no sentido de expressar o comportamento reativo destes sistemas de forma mais clara e real. Dentre estes trabalhos, destaca-se *Statecharts* [1], uma proposta de linguagem formal e visual para especificação de sistemas reativos complexos. Esta linguagem obteve grande sucesso na comunidade de desenvolvedores de software, sendo inclusive referência para modelagem de controle na difundida *Unified Modeling Language* (UML) [2]. A complexidade na descrição de um sistema reativo distribuído pode ser fruto dos requisitos gerais de concorrência e distribuição da solução final o que tornam a tarefa de especificação destes sistemas ainda mais difícil [6]. O projeto de sistemas reativos concorrentes e distribuídos é uma área que tem sido tema de alguns outros trabalhos como [5] e será o objeto principal deste trabalho como detalhado a seguir.

Sistemas reativos também estão presentes no contexto de desenvolvimento de interfaces de usuário. A interface de usuário (interface, por simplicidade) é o componente de um sistema interativo responsável por traduzir as ações do usuário em ativações das funcionalidades da aplicação é também responsável por exibir os resultados destas interações. O sub-componente da interface que captura esta sintaxe de interação é comumente conhecido como gerenciador de diálogo da interface e o mesmo pode ser classificado como um sistema reativo. A tecnologia Xchart [3] é uma proposta que foi originalmente criada para reduzir as dificuldades de desenvolvimento de gerenciadores de diálogos de interfaces complexas, ao contrário de várias outras propostas que dão ênfase ao componente de apresentação. Xchart é uma variante de *Statecharts* e pode muito bem ser utilizada em um domínio mais amplo de sistemas reativos, o nicho de sistemas reativos distribuídos.

A tecnologia Xchart propõe uma linguagem visual de mesmo nome (Linguagem Xchart) e um ambiente de apoio para especificação e execução de sistemas reativos complexos. A especificação de comportamento reativo por meio da linguagem Xchart permite expressar a execução concorrente de elementos de controle e, portanto, a sua potencial distribuição por diversos pontos de uma rede. O ambiente Xchart oferece ferramentas para edição e execução de especificações descritas em Xchart. O projetista de um componente reativo (ou componente de controle) neste ambiente definirá o comportamento desejado através de diagramas na linguagem Xchart. Estes diagramas oferecem uma forma de expressão de maior nível de abstração se comparado a linguagens de programação como, por exemplo, Java ou a linguagem C.

Diagramas descritos em Xchart podem ser executados de forma concorrente e distribuída por uma rede de computadores. A questão relativa ao fato destes diagramas estarem ou não distribuídos em tempo de execução deve ser totalmente transparente ao projetista de um sistema no ambiente Xchart. O projetista não precisa se preocupar em detalhes como localização de recursos, controle de concorrência, chamada de procedimentos remotos, e outras necessidades de mais baixo nível de uma aplicação distribuída. Diagramas descritos em Xchart em tempo de execução são chamados de instâncias.

O ambiente de execução Xchart divide-se em dois subsistemas básicos: o Servidor Xchart e o Gerente de Distribuição. O Servidor Xchart (**SX**) é responsável pela execução propriamente dita das especificações, ele produz a reação de uma instância de Xchart de acordo com os estímulos externos e a sua atual configuração conforme a semântica formal de Xchart. O Gerente de Distribuição (**GD**) é o componente que oferece os recursos de distribuição para a execução das especificações de controle descritas em Xchart. O **SX** eventualmente requisita serviços do **GD** para a execução distribuída. O **GD** tem o papel de integrar as diversas instâncias de **SX** em um ambiente distribuído.

O trabalho de mestrado [4] teve como um de seus resultados, a implementação de um protótipo do ambiente de execução Xchart totalmente desenvolvido na linguagem Java. O desenvolvimento do **GD** foi baseado em tecnologias para sistemas distribuídos disponíveis na plataforma J2EE. Esta estratégia teve o intuito de reduzir os esforços relativos à programação das primitivas de distribuição de mais baixo nível, além de fornecer uma solução que fosse independente de plataforma e hardware.

O objetivo do trabalho nesta fase de doutorado é o de apresentar a tecnologia Xchart como uma alternativa real para a especificação e execução de sistemas reativos distribuídos. O foco estará na elaboração completa dos requisitos para um ambiente de especificação e execução de software com controle distribuído. Este ambiente terá entre suas atribuições, a especificação, execução, simulação, testes e documentação de sistemas reativos distribuídos. A semântica de Xchart deverá ser revista para atender de forma mais eficiente estas atribuições. Nota-se que muitas das propostas que abordam a distribuição de sistemas reativos, deixam a cargo do projetista a responsabilidade por codificar parte do trabalho de distribuição [5] [7]. Xchart diferencia-se das propostas atuais elaboradas com base em *Statecharts*, pois a sua semântica se comparada com a de *Statecharts* é mais completa para especificação de software de controle distribuído. Xchart prevê mecanismos de controle de concorrência no acesso a recursos compartilhados (variáveis de controle globais) e também serviço de entrega de eventos para instâncias de controle distribuídas (*broadcast*, *multicast* e ponto-a-ponto).

O requisito de transparência quanto à distribuição para o projetista do sistema, continuará sendo uma das características deste trabalho de forma a não sobrecarregar a tarefa de especificação. Os requisitos do ambiente nesta fase também passam pela discussão sobre segurança e tolerância a falhas. Os algoritmos distribuídos utilizados anteriormente serão revistos e soluções mais eficientes serão propostas. A

decisão quanto as tecnologias que serão utilizadas neste projeto serão tomadas no decorrer do trabalho, porém hoje tem-se em mente a necessidade de abordar novas demandas como SOA (*Service Oriented Architecture*) para permitir uma melhor interoperabilidade dos componentes que fazem parte da solução. Maiores detalhes sobre o andamento do projeto Xchart podem ser consultados no site oficial do projeto [8].

Referências

- [1] David Harel. *Statecharts: A Visual Formalism for Complex Systems*. *Science of Computer Programming*, 8(3): 231-274, June 1987.
- [2] Booch, Rumbaugh, and Jacobson. *The Unified Modeling Language User Guide*. Addison-Wesley, 1998.
- [3] Fábio Nogueira de Lucena e Hans Kurt Edmund Liesenberg. Um Modelo de Especificação e Implementação de Gerenciadores de Diálogo. *Tese de Doutorado, IC/UNICAMP, Campinas/SP, Dezembro 1997*.
- [4] Carlos Neves Júnior e Hans Kurt Edmund Liesenberg. Gerente de Distribuição do Ambiente Xchart em J2EE: Distribuindo Sistemas Reativos com a Tecnologia Xchart. *Tese de Mestrado, IC/UNICAMP, Campinas/SP, Setembro 2005*.
- [5] Hassan Gomaa. *Designing Concurrent, Distributed Real-Time Applications with UML*. Addison-Wesley, 2000.
- [6] Pnueli, Amir e Rosner, Roni. Distributed Reactive Systems are Hard to Synthesize. *Proceedings of 31st IEEE Symp. on Foundations of Computer Science*. 1990. pp. 746-757.
- [7] Mark Mahoney and Tzilla Elrad. Distributing Statecharts to Handle Pervasive Crosscutting Concerns. *Building Software for Pervasive Computing Workshop at OOPSLA, October 2005*.
- [8] Site do Projeto Xchart. <http://xchart.incubadora.fapesp.br> Novembro 2006.

6. A framework based on semantic Web services and AI planning for the management of bioinformatics scientific workflows

Authors: Luciano Antonio Digiampietri, João C. Setubal, and Claudia B. Medeiros

INTRODUCTION Bioinformatics activities present new challenges, such as how to exchange and reuse successful experimental procedures (programs and parameters), tools and data, and how to understand and provide interoperability among data and tools across different sites, for distinct user profiles.

Scientific workflows [7] are being increasingly adopted as a means to specify and coordinate the execution of experiments that involve participants in distinct sites. Such workflows allow the representation and support of complex tasks that use heterogeneous data and software. They differ from business workflows in several points. In particular, in bioinformatics they are characterized by a high degree of human intervention and variability in workflow design for the same task (e.g., there are several ways to assemble a genome).

Scientific workflows are usually designed manually. Manual composition is a hard work and susceptible to errors. Automatic composition of Web services is a recent trend in Artificial Intelligence [2]. Users should be able to specify “what” they desire (high level goals), and the system supplies the “how” - the Web services to be used, how to interact with those services, etc. The task of presenting a sequence of actions to achieve an objective is called in AI *planning* [6]. Many planning algorithms can be considered (e.g., symbolic model checking and hierarchical planning). For a detailed comparison of planning techniques, the reader is referred to [1], where we justify our choice of SHOP2 [4] to support automatic design of scientific workflows.

GOALS The goals of this thesis are to specify and prototype a framework that allows the integration of heterogeneous data and tools and the design of experiments as scientific workflows. Moreover, it takes advantage of AI planning to support automatic composition of tasks. It also uses ontology repositories for enhancing plan generation and to facilitate data and tool annotation, thus helping traceability.

OUR PROPOSAL Our approach extends the current planners by treating complex objects and objects created dynamically, two very important characteristics in Web services, not usually found in AI planning. Moreover our workflow design and execution process is based on combining AI planning with information stored in three repositories to improve plan generation: Ontology Repository, Service Catalog and Workflow Repository.

The **Ontology Repository** contains a domain and a service ontologies that will be used to support automatic composition and annotation of services and workflows.

The **Service Catalog** plays the role of a UDDI (Universal Description Discovery & Integration) registry, enhanced with extended functionalities. Each service entry in the Catalog is annotated with the ontologic concepts of the Service Ontology. The **Workflow Repository**, adopted from WOODSS, stores annotated (sub)workflows at different abstraction levels [3]. All the repositories are interrelated.

Our architecture is able to deal with automatic composition of workflows based in Web services. Figure 1 shows this architecture, highlighting the main modules and their interactions. It combines basic features of a planning framework (e.g., such as that in [5]) with the scientific workflow framework of WOODSS [3], adding catalog and user interaction facilities. Whereas WOODSS is based on manual composition, our architecture supports automatic and semi-automatic composition via planning. For a detailed description of the architecture modules, the reader is referred to [1].

We implemented a prototype of our architecture to solve two important bioinformatics problems: genome assembly and annotation. Both problems can be solved by invoking a sequence of specialized tools, already available at several sites, in distinct flavors. The issue is to construct the appropriate workflow.

This requires expert knowledge (e.g., to choose tools among alternatives) and domain knowledge (captured by ontologies). We have specified such an ontology which extends TAMBIS (www.tambis.org). We also created an annotated service catalog, containing information about commonly used bioinformatics tools on the Web. Given all this information, the planner supports automatic or manual workflow design. For more details on our case study the reader is referred to [1].

CONCLUSIONS This thesis presents a solution to the problem of specifying a scientific workflow to execute on the Web. Our main contributions lie in proposing and prototyping a framework that takes advantage of AI planning techniques, combined with ontologies and Semantic Web standards to support workflow design. The solution is based on repositories that store information on services, ontologies, and workflows. In particular, ontology repositories are extensively used in enhancing plan generation with semantics, in helping users design better scientific workflows, and in helping traceability.

Our architecture is generic, and can be instantiated for several domains. It helps the user in the three kinds of composition: manual (supervised), iterative and automatic. Manual composition is useful when the user knows exactly what activities he/she desires to compose. Iterative composition is advisable when the user has a general knowledge of the process that he/she wants to execute, but does not know what tools/services execute this process. Automatic composition is advisable when the user knows pre- and post-conditions, but does not know (or is not interested in) how to design a workflow that satisfies these conditions.

STATUS OF WORK The architecture and workflow model were specified. Present implementation supports manual and automatic workflow design, where the

planner extends SHOP2. Ongoing work concerns in improving provenance and traceability support. The next steps are related to workflow execution.

ACKNOWLEDGMENTS This thesis was partially financed by FAPESP, CAPES, CNPq and Microsoft Research. This work was developed in cooperation with J. J. Pérez-Alcázar from EACH, USP.

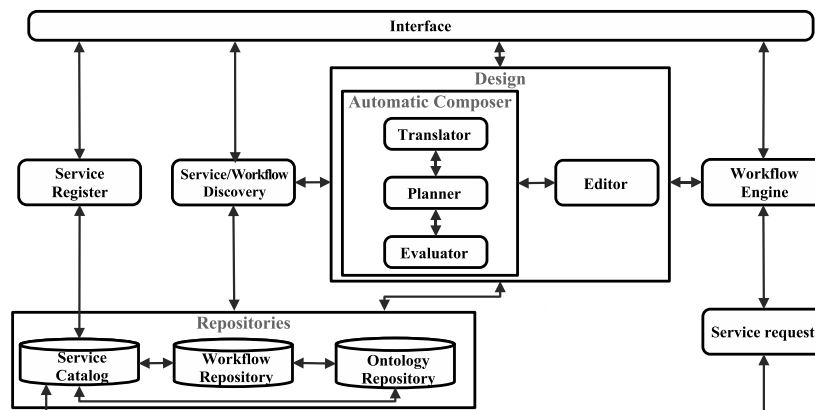


Figura 1: System Architecture

Referências

- [1] L. Digiampietri, J. Pérez-Alcazar, C. Medeiros, and J. Setubal. A framework based on semantic Web services and AI planning for the management of bioinformatics scientific workflows. Technical Report IC-06-004.
- [2] *Workshop on Planning and Scheduling for Web and Grid Services*, June 2004.
- [3] C. B. Medeiros, J. Perez-Alcazar, L. Digiampietri, G. Pastorello, A. Santanche, R. Torres, E. Madeira, and E. Bacarin. WOODSS and the Web: Annotating and Reusing Scientific Workflow. *ACM SIGMOD Record*, 34(3):18–23, 2005.
- [4] D. Nau, T. C. Au, O. Ilghami, U. Kuter, W. Murdock, D. Wu, and F. Yaman. SHOP2: An HTN Planning System. *Journal of Artificial Intelligence Research*, 20:379–404, 2003.
- [5] J. Rao and X. Su. A Survey of Automated Web Service Composition Methods. In *SWSWPC 2004*, volume 3387, pages 43–54, 2004.
- [6] S. Russel and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 2003.
- [7] J. Wainer, M. Weske, G. Vossen, and C. B. Medeiros. Scientific Workflow Systems. In *NSF Workshop on Workflow and Process Automation Information Systems*, 1996.

7. Uma abordagem em camadas para especificação de políticas de segurança de sistemas operacionais para ambientes heterogêneos

Autores: Diogo Ditzel Kropiwiec, Paulo Lício de Geus

O gerenciamento de segurança de sistemas computacionais está cada vez mais complexo. À medida que as redes de computadores são interligadas através da Internet, torna-se necessária a integração de uma variedade de mecanismos para garantir os requisitos de segurança das redes de organizações e de computadores pessoais [1]. Entre os mecanismos utilizados, sistemas operacionais seguros são essenciais à segurança de sistemas computacionais, pois permitem restringir o acesso de aplicações aos recursos do sistema, limitando significativamente o efeito sobre o sistema de um ataque bem sucedido a uma aplicação [2].

Para operar corretamente, os sistemas operacionais seguros requerem que suas políticas de segurança sejam definidas de forma a refletir os requisitos de proteção necessários às organizações e computadores pessoais [3]. Entretanto, os modelos de políticas de segurança mais utilizados em sistemas operacionais são baseados em dois contextos distintos: modelos de controle de acesso do sistema operacional, como o *ACL* (*Access Control List*) [4] e o *DTE* (*Domain and Type Enforcement*) [5]; e modelos focados em aplicações, como o *RBAC* (*Role Based Access Control*) [6] e *MLS* (*Multi-Level Security*) [7]. Em consequência disso, a especificação de uma política de segurança que reflita corretamente os requisitos de segurança é um processo complexo: tanto aspectos de alto-nível (focados em usuários e atividades) quanto aspectos de baixo-nível (focados em processos, arquivos e recursos) devem ser considerados, porém os modelos privilegiam apenas um determinado nível.

O problema apresentado torna-se ainda maior ao se considerar que a maioria das organizações dispõe de redes de computadores heterogêneas [1]. A consequente necessidade de utilizar sistemas operacionais distintos dificulta o processo de gerenciamento das políticas de segurança. Mesmo aqueles que implementam o mesmo modelo podem utilizar linguagens de especificação distintas (por exemplo, [8] e [9] usam especificações distintas para *RBAC*), inviabilizando uma verificação de que as várias especificações em linguagens e modelos diferentes refletem corretamente os mesmos requisitos de segurança originais

Tendo em vista o cenário apresentado, este trabalho pretende desenvolver uma sistemática para o gerenciamento da configuração de políticas de segurança de sistemas operacionais, com foco especial às necessidades dos ambientes de redes heterogêneas encontrados nas organizações atuais. A abordagem, que está sendo desenvolvida segundo o paradigma de Gerenciamento Baseado em Modelos (*MBM - Model-Based Management* [10]), utilizará uma estrutura dividida em níveis de abstração, na qual o nível mais alto representa os requisitos de segurança com relação à visão organizacional (e, portanto, independente de plataforma), e o nível mais baixo representa as regras relacionadas à visão técnica da especificação da política.

Utilizando essa estrutura, o processo de especificação de uma política de segurança poderá ser feita de maneira controlada e verificável, partindo da visão organizacional até atingir a visão técnica. O detalhamento de cada nível do modelo será obtido a partir do refinamento do nível imediatamente superior, permitindo acompanhar a transformação dos requisitos de segurança em regras de uma política específica. No contexto de mecanismos de segurança de redes [11], a automatização do processo de refinamento entre níveis e a verificação de corretude com relação ao nível anterior foi empregado com sucesso para aumentar o grau de confiança na política especificada utilizando o *MBM*, garantia que poderá ser obtida no contexto de sistemas operacionais.

O método empregado para o desenvolvimento desse trabalho consiste em três etapas principais. Primeiro, será realizado o estudo de modelos de segurança, com ênfase em modelos derivados de *MBM* e em modelos aplicados a sistemas operacionais, necessário para o desenvolvimento da abordagem proposta. Segundo, serão estudadas as contribuições na modelagem de segurança no contexto de mecanismos de segurança de redes utilizando o paradigma de *MBM*, de forma a expandí-las para o contexto de sistemas operacionais. Dessa forma, características como escalabilidade, formalização do processo de refinamento e ferramentas auxiliares poderão ser incorporadas a este trabalho. Finalmente, o modelo desenvolvido será utilizado para especificar uma política de um sistema operacional para um conjunto de cenários, de forma a validar o modelo.

Os resultados pretendidos nesse trabalho são: (i) desenvolvimento de um modelo de especificação de políticas de segurança para sistemas operacionais que compreenda tanto aspectos de alto-nível quanto aspectos de baixo nível; (ii) desenvolvimento de um processo de especificação que permita uma transformação gradual e controlada a partir da visão organizacional até a visão técnica, garantindo que os objetivos de segurança estão corretamente especificados em cada nível e a dissociação das duas visões.

O trabalho ainda encontra-se nos estágios iniciais de desenvolvimento, porém a proposta é potencialmente promissora, tendo em vista os resultados obtidos com o modelo no contexto de mecanismos de segurança de redes. Diante do complexo processo de estabelecer e manter políticas de segurança, este trabalho pretende oferecer contribuições que melhorem o gerenciamento de segurança de sistemas operacionais, expandindo o uso do paradigma de *MBM*.

Referências

- [1] E. Nakamura and P. L. de Geus. *Segurança de Redes em ambientes cooperativos*. Editora Berkeley, São Paulo, first edition, 2002.
- [2] D. D. Kropiwiec and P. L. de Geus. Paradigmas de segurança em sistemas operacionais. In *Anais do IV Workshop de Segurança em Sistemas Computacionais*, Gramado/RS, Maio 2004.

- [3] M. Bishop *Computer Security: Art and Science*. Addison-Wesley, 4th edition, November 2003.
- [4] A. S. Tanenbaum and A. S. WoodHull. *Operating Systems: Design and Implementation*. Prentice Hall, Upper Saddle River, New Jersey, USA, 2nd edition edition, 1997.
- [5] K. W. Walker, D. F. Bagder, M. J. Petkac, L. Sherman, and K. A. Oostendorp. Confining root programs with domain and type enforcement. In *Proceedings of The 6th USENIX Security Symposium*, San Jose, California, 1996.
- [6] R. Sandhu, D. Ferraiolo, and R. Kuhn. The NIST model for role-based access control: Towards a unified standard. In *Proceedings of the 5th ACM Workshop on Role-Based Access Control*, pages 47 – 63, Berlin, Germany, 2000. ACM Press, New York, NY, USA.
- [7] D. E. Bell and L. J. La Padula. Secure computer systems: Mathematical foundations and model. Technical Report MTR-2547 Vol I, MITRE Corporation, Bedford, MA, EUA, March 1973.
- [8] S. D. Smalley. Configuring the SELinux policy. Technical report, National Security Agency of United States of America, January 2003.
- [9] J. Hoffman Implementing RBAC on a type enforced system In *13th Annual Computer Security Applications Conference (ACSAC '97)*, Roseville, MN, USA, 1997.
- [10] I. Lück and M. Schönbach and A. Mester and H.Krumm. Derivation of backup service management applications from service and system model. In *R. Stadle, B.S., ed.: Active Technologies for Network and Service Management, Proc. DSOM'99q*. Number 1700 in Lecture Notes in Computer Science, Heidelberg, Springer Verlag (1999).
- [11] J. Porto de Albuquerque and H. Krumm and P. L. de Geus Policy modeling and refinement for network security systems. In *6th IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY 2005)*, pages 24-33, Stockholm, Sweden, June, 2005.

8. Dynamic Traffic Grooming in IP over WDM Networks with Quality of Service

Authors: André Costa Drummond and Nelson L. S. da Fonseca

In the near future the establishment of optical Wavelength Division Multiplexing (WDM) networks will give birth to a new era of virtually endless bandwidth. In this technology the data transport channel is called lightpath and it uses only one wavelength in one fiber optic link. WDM is considered the switching technology for the new generation of optical Internet backbone, since it provides massive bandwidth availability, but at the same time it requires fast switching at the Internet Protocol (IP) level. Multiprotocol label switching (MPLS) has been deployed in Internet Protocol (IP) backbones to meet the switching needs as well as to support quality of service provisioning. Moreover, the Generalized MPLS (GMPLS) allows wavelength switching besides those supported by MPLS, thus, facilitating the integration of IP and optical control planes [1].

Considering that each lightpath has several gigabit per second of bandwidth capacity, the transport of low speed flows through the network will certainly lead to network resource wastage. The solution to this problem lies on the aggregation (grooming) of those flows in order to better use the available lightpath capacity. Every time that a flow is added or removed from a lightpath an electronic switch operation is required which implies that highly expensive switching equipment has to be used. The traffic grooming problem is defined as the problem of grooming low speed traffic flows and transport those through established lightpaths minimizing the total network cost with optical-electronic-optical switching equipment.

The bandwidth demands of traffic flows in backbone networks are usually considered static due to the high level of traffic aggregation in these networks. However, in Metropolitan Area Networks (MANs), such levels of aggregation are not observed; therefore, there is a need to take into account the fluctuations of bandwidth demand, leading to the so called Dynamic traffic grooming problem.

Quality of service provisioning is one of the main issues in the integration of IP and optical control planes. Nonetheless, most previous research has proposed solutions which are too simplistic, involving non-realistic assumptions as well great computational complexity [2]. The aim of the present proposal is the development of effective and efficient techniques for the dynamic grooming of subwavelength LSPs into a lightpath without ignoring QoS requirements. Traffic grooming algorithms work at edge IP nodes to decide in which lightpaths IP flows should be transported. GMPLS will be used for interworking the IP and the optical domains.

Our proposal focus on the development of a solution for the dynamic traffic grooming problem that takes into consideration the QoS requirements of the IP layer and the physical impairments of the optical layer [3]. The network scenario consider in our work will be the mesh metropolitan area networks and the performance evaluations will be carried out with realistic dynamic traffic models.

In our attempt to develop useful mathematical models we shall consider the following situations:

- (i) multi-hop virtual topology design, which includes lightpaths that can aggregate/disaggregate traffic at any convenient point along the path, not only at the source or destination nodes;
- (ii) arbitrary mesh topologies;
- (iii) multi-rate traffic models, since diverse traffic granularity more accurately portrays real network traffic;
- (iv) heterogeneous resources, i.e. networks composed of different types of equipment, including optical cross-connects (OXC) with limited switching and conversion capabilities.

The grooming mechanisms developed will be coupled with the following QoS provisioning mechanisms:

QoS mapping. This process maps the QoS requirements of IP flows into lightpaths. It defines classes of services, both for the IP layer and for the optical layer, as well as mapping strategies. The mapping process involves:

- (i) Traffic and Lightpath classification. This process considers IP flow requirements and lightpath capabilities, classifying these into classes of service.
- (ii) Traffic and Lightpath mapping. This process maps traffic classes into lightpaths classes, defining which lightpath should be used to carry a certain type of traffic.
- (iii) Traffic scheduling and grooming. This mechanism determines the flows to be groomed.

QoS routing. This mechanism uses constraint-based routing (CBR) schemes, in which constraints not only consider on classical traffic grooming issues, such as minimization of the use of o-e-o equipment, but also the QoS requirements of LSPs. It is designed to consider physical impairments of a lightpath when defining the route for a specific LSP.

Approximation algorithms will be used to find solutions with low computational complexity that can be executed in real-time.

Referências

- [1] Jing Fang and Arun K. Somani. IP traffic grooming over WDM optical networks. In *Proceedings of IFIP/IEEE 9th conference on Optical Network Design and Modeling*, Milan, Italy, February 2005.

- [2] Shu Huang and Rudra Dutta. Research problems in dynamic traffic grooming in optical networks. In *Proceedings of BROADNETS*, San José, California, 2004.
- [3] Wei Wei, Zhongheng Ji, Junjie Yang, and Qingji Zeng. On integrated QoS control in IP/WDM networks. In *Proceedings of ICN*, pages 334–341, 2005.

9. O método da geração de colunas aplicado ao problema do separador de vértices em grafos

Autores: Edna Ayako Hoshino e Cid C. de Souza

Dado um grafo conexo não-orientado $G = (V, E)$, um inteiro $b \leq |V|$ e um custo inteiro $c_i \geq 0$ associado a cada vértice i em V , o **problema do separador de vértices**, ou *vertex separator problem* (VSP), consiste em encontrar uma partição de V em subconjuntos não-vazios A , B e C tal que (i) os vértices em A não sejam adjacentes a vértices em B , (ii) $\max\{|A|, |B|\} \leq b$ e (iii) $\sum(c_i : i \in C)$ seja mínima quando sujeito a (i) e (ii). Os subconjuntos A e B são chamados **shore** e C é denominado **separador**.

O problema do separador de vértices é NP-difícil. Balas e de Souza [BdS05a] estudaram algumas classes de desigualdades válidas e as condições em que elas definem facetas para o poliedro do VSP. Resultados computacionais da aplicação delas em um algoritmo *branch-and-cut*, também, são reportados [BdS05b]. Borndörfer, Ferreira e Martin [BFM98], também, propuseram um algoritmo *branch-and-cut*, mas com uma família diferente de cortes e para um problema mais geral que o VSP, que procura por um separador para um número fixo β de *shores* (o VSP é um caso especial em que $\beta = 2$). Este resumo reporta parte dos resultados computacionais da aplicação do método da geração de colunas para a resolução do VSP. Não temos o conhecimento de nenhum trabalho anterior propondo a resolução do VSP por este método. Mais do que avaliar o desempenho de um algoritmo de geração de colunas na resolução do VSP, o objetivo deste estudo preliminar consiste em avaliar o uso de algumas técnicas para acelerar a convergência e lidar com problemas como degenerescência e instabilidade dos duais, que surgem na utilização deste método.

As decomposições de Dantzig-Wolfe [DW60] e de mestre explícito [dAU03] foram utilizadas para obter uma formulação extensiva do problema. Nos testes realizados, os limitantes duais obtidos com estas decomposições foram mais apertados que aqueles dados pela relaxação linear da formulação compacta (cerca de 10% mais apertado e, em algumas instâncias, igualaram-se à solução ótima). Isso decorre da decomposição escolhida, cujo problema de *pricing* consiste em gerar um *shore* e sua vizinhança no separador. O tempo de processamento gasto na resolução deste problema de *pricing* foi consideravelmente alto, o que era esperado, dada a alta qualidade do limitante dual. Este é um famoso dilema em métodos de geração de colunas.

Embora o método da geração de colunas, utilizando-se a decomposição escolhida neste estudo, não se mostrou adequado ao VSP, testes foram realizados sobre algumas instâncias pequenas (instâncias MM-I em [BdS05b]) com o intuito de avaliar as diferentes técnicas para acelerar o processo de geração de colunas. A utilização de um *pool* de colunas mostrou-se bastante efetivo, reduzindo, em média, 50% do tempo de processamento. Uma redução de cerca de 20%, em média, no tempo de processamento foi obtida com a utilização dos tetos dos limitantes duais [Las70, VW96]. A

utilização de *pricing* parcial, com uma heurística GRASP, conseguiu ter um ganho de até 5 vezes no desempenho em relação ao *pricing* exato. Um fato interessante é que a heurística falhou um número muito pequeno de vezes e os limitantes duais obtidos nestes casos foram apertados o suficiente para encerrar o processo de geração de colunas com poucas iterações. O uso de uma heurística primal melhorou muito pouco o desempenho do algoritmo de geração de colunas, no entanto, os resultados mostram que as colunas geradas pela heurística primal foram utilizadas na maioria das soluções ótimas do problema mestre.

Um dos problemas que ocorrem em algoritmos de geração de colunas refere-se à grande oscilação dos duais durante a geração de colunas, que pode dificultar a convergência do método. Para resolver este problema, dois métodos de estabilização foram analisados. Os resultados da utilização destes métodos não foram expressivos. A dificuldade na utilização destes métodos está na escolha adequada dos parâmetros e das estratégias de atualização dos mesmos. O método *box stabilization* [dMVDH99] reduziu relativamente pouco o tempo de processamento, mas conseguiu resolver o problema de degenerescência que ocorre nos problemas mestres. Outra técnica utilizada para reduzir a degenerescência foi a perturbação do *rhs* das restrições. Esta técnica mostrou-se efetiva na remoção da degenerescência.

Diferentes combinações de estratégias foram implementadas para avaliar a melhor implementação para o algoritmo de geração de colunas no VSP. A Tabela 1 resume a média de tempo (**mTime**), tempos mínimos (**minTime**) e máximos (**maxTime**), média de *gap* (**mGap**), *gap* mínimo (**minGap**) e máximo (**maxGap**) nas diferentes combinações de implementação.

codigo	mTime	minTime	maxTime	mGap	minGap	maxGap
DW	649.90	13.29	1805.44	0.01	0.00	0.06
DW-h	644.56	17.16	1800.09	0.01	0.00	0.06
DW-p	369.71	14.08	1799.73	0.00	0.00	0.03
DW-hp	342.38	6.68	1799.97	0.00	0.00	0.03
DW-hsp	338.43	7.64	1799.52	0.01	0.00	0.04
Grasp	328.16	1.53	1799.13	0.00	0.00	0.02
Grasp-h	302.75	0.85	1799.56	0.00	0.00	0.01
Grasp-p	253.88	1.94	1799.27	0.00	0.00	0.01
Grasp-hp	255.79	0.54	1799.80	0.00	0.00	0.01
Grasp-hsp	184.37	0.50	1239.59	0.00	0.00	0.00

Tabela 1: Resumo comparativo das implementações nas instâncias MM-I.

Nesta tabela, a implementação de geração de colunas clássica é denotada por DW e a versão com *pricing* parcial por Grasp. O sufixo -h é usado para indicar a utilização de heurística primal, enquanto -p para indicar a versão com perturbação no *rhs*. O sufixo -hsp indica que tanto a heurística, quanto o método de estabilização e perturbação no *rhs* foram utilizados. Pode-se notar que a versão que utiliza heurística primal, *pricing* parcial, método de estabilização e perturbação do

rhs consegue reduzir em até 70% o tempo de processamento em relação ao algoritmo de geração de colunas clássico.

Embora o método da geração de colunas não tenha apresentado resultados satisfatórios para o VSP, as técnicas estudadas mostram que elas podem ser combinadas para reduzir significativamente o desempenho de algoritmos de geração de colunas.

Referências

- [BdS05a] E. Balas e C. C. de Souza. The vertex separator problem: a polyhedral investigation. *Mathematical Programming*, 103(3):583–608, 2005.
- [BdS05b] E. Balas e C. C. de Souza. The vertex separator problem: algorithms and computations. *Mathematical Programming*, 103(3):609–631, 2005.
- [BFM98] R. Bordörfer, C.E. Ferreira, e A. Martin. Decomposing matrices into blocks. *SIAM Journal on Optimization*, 9:236–269, 1998.
- [dAU03] M. Poggi de Aragão e E. Uchoa. Integer program reformulation for robust branch-and-cut-and-price, August 2003. Working paper. Presented at ISMP’03. Copenhagen.
- [dMVDH99] O. du Merle, D. Villeneuve, J. Desrosiers, e P. Hansen. Stabilized column generation. *Discrete Mathematics*, 194:229–237, 1999.
- [DW60] G. Dantzig e P. Wolfe. Decomposition principle for linear programs. *Operations Research*, 8:101–111, 1960.
- [Las70] L.S. Lasdon. *Optimization Theory for Large Systems*. MacMillan, 1970.
- [VW96] François Vanderbeck e Laurence A. Wolsey. An exact algorithm for ip column generation. *Operations Research Letters*, 19:151–159, 1996.

10. Contract Negotiation on Agricultural Supply Chains

Authors: Evandro Bacarin, Edmundo R.M. Madeira, and Claudia Bauzer Medeiros

Introduction

Trading relations inside a supply chain comprise a huge amount of commercial transactions and are subject to legal commitments. The increasing use of computational means to perform commercial transactions demands the replacement of contracts by their electronic counterparts, and performing live negotiation through software agents. Electronic contracts and electronic negotiation are often researched as problems in computer networks. However, they also raise interesting database problems, such as: (i) the negotiators must agree on a mutually known set of concepts and names, (ii) contracts (and their versions) must be recorded and their execution must be tracked in order to check if partners have fulfilled (or are fulfilling) their duties (audit), (iii) audit procedures may have to take into account temporal and spatial constraints mentioned in a contract, (iv) a negotiation process may raise a number of inter-related contracts, demanding special care to enforce consistency among them in case the agreement is renegotiated. Our proposal encompasses a negotiation framework focusing on agricultural supply chains that combines research on computer networks and databases to try to solve these four issues.

Objectives

The objectives of this proposal are to: i) establish a model to support integration of agricultural supply chains; ii) design a negotiation protocol that allows multi-partner negotiation, including quota negotiation; iii) define an electronic contract structure that is subject to the constraints focused in agricultural supply chains; iv) design of a negotiation (renegotiation) framework based on Web services and implementation of its main parts; v) design of an enactment infrastructure and implementation of its main parts.

Methodology

The methodology followed is described under the goals.

The model and the architecture. First, we have to create a model that defines an agricultural supply chain. This allows us to know the actors within this kind of supply chain, the role they play, their interactions, how they are organized, and the external factors that have influence inside the supply chain. Based on this model, we can propose an architecture that aims at integrating agricultural supply

chains and comprises several activities, including: coordinating the activities inside a supply chain, registering the execution of such activities, enforcing quality criteria on the product flow, negotiating and executing contracts among supply chain partners. This architecture encompasses a number of repositories, including one for contracts. Each specific repository may be distributed, replicated, may have different versions and its contents may have to address spatial and temporal issues.

The contract. Second, our research emphasis is on contract specification, negotiation and execution. Other ongoing researches at IC-UNICAMP deal with other activities, namely coordination and registering. We plan to define what is a contract in an agricultural supply chain, including the definition of its syntax and how it should be executed.

The negotiation protocol. Third, we have to design a negotiation protocol based on exchange of asynchronous messages that can produce such a contract or can renegotiate an existing one. The messages would include an endless number of attributes (product names, quality designators, currencies, etc) that must be understood by all negotiators. Thus, ontologies should be used.

Negotiation strategies. Fourth, the negotiation protocol should allow a number of distinct negotiation strategies. Some negotiators may use AI techniques, while others may use a game theory approach. Yet, others may have a strategy based on statistical or optimization techniques. Thus, we plan to study some strategies and implement a prototype for a couple of them to assess the flexibility of our negotiation protocol. Clearly, there is a great number of distinct strategies, each of which may lead to other research activities.

Types of agreements. Fifth, the negotiation protocol should also be suitable for reaching several types of agreements, in special, the negotiation of quotas distributed over a set of negotiators. Quota assignment, in general, is constrained by criteria, for instance, no negotiator should be granted a quota 10% higher than the quota of the negotiator that received the smallest grant. We intend to direct the implementation of the previous step towards quota negotiation.

The negotiation framework. Sixth, we plan to implement a prototype of a framework for negotiation, based on Web services, including a middleware that supports conversation among the negotiators and auxiliary objects. The implementation will use Java. So far, we have written the main interfaces needed by a negotiator in Java, and their dynamics shown as interaction diagrams.

We intend to implement the prototype negotiators in two phases: standalone and as Web services. In the first phase, we evaluate the negotiation protocol without all the complexity added by a Web service layer. The messages exchanged among the negotiators will boil down to method calls. Next, we will add that layer over the negotiators. The messages will be mapped to SOAP messages. The interfaces will be mapped to Web services interfaces and described using WSDL. In addition, many negotiators may take part in a negotiation. Thus messages may be broadcasted. We have to investigate how this can be performed using Web services.

Results

We have completed most of the design phase: i) The model and the architecture were published in [1]. ii) We have designed an initial version for the contract structure. iii) We have designed the messages that are needed to allow all negotiation scenarios we are interested in. We also have described the order in which they should be exchanged to be meaningful. This was done using a context-free grammar. The designed grammar only shows possible sequences of message exchanges, but it does not make clear the steps a negotiator should follow after it sends a specific message and the steps the negotiator that received it should follow. This is done by means of state diagrams (UML). The negotiation process is conducted by a leader negotiator, guided by a contract model, and allows several styles of negotiation (e.g., bargains, auctions, ballots). iv) The interfaces a negotiator should implement to receive the negotiation messages were written in Java. We identified 20 interfaces with a total of 80 methods. Those interfaces and a number of classes are the main part of our negotiation framework. In order to assess the applicability of this framework, we have written interaction diagrams for typical negotiation patterns.

Conclusions

The objective of our research is to design and develop a framework for contract negotiation and execution in the context of agricultural supply chains. So far, we have done most of the design activities. The implementation of the main parts of the negotiation framework are at the very beginning steps. There are a number of interesting questions to be addressed during the research development or after thesis's conclusion. They include: usage of ontologies within the negotiation and contract execution; implementation of distinct negotiation strategies (e.g., using game-theoretic approaches, AI techniques, heuristics), and verification whether the contract has been fulfilled.

Contract execution can be seen as a workflow execution. In this context contract fulfillment can take advantage of workflow mining result, such as: [2], [3].

Referências

- [1] E. Bacarin, C.B. Medeiros, and E.R.M. Madeira. A Collaborative Model for Agricultural Supply Chains. In *CoopIS 2004, LNCS 3290*, pages 319–336, 2004.
- [2] W.M.P. van der Aalst, H.T. de Beer, and B.F. van Dongen. Process mining and verification of properties: An approach based on temporal logic. In *OTM Conferences (1)*, pages 130–147, 2005.
- [3] W.M.P. van der Aalst, B.F. van Dongen, J. Herbst, L. Maruster, G. Schimm, and A.J.M.M. Weijters. Workflow mining: A survey of issues and approaches. *Data & Knowledge Engineering*, 47(2):237–267, 2003.

11. Detecção de anomalias em *logs* de sistemas de apoio a processos de negócios

Autores: Fábio de Lima Bezerra e Jacques Wainer

A mineração de processos (*Process mining*) é uma técnica que visa construir um modelo de processo (ou workflow) a partir de um histórico de execução de instâncias de processos, também chamado de *log* [1–4]. O histórico de execução de uma instância em particular é chamado de *trace*, formado pela lista das atividades executadas pela instância. As atividades que formam o *trace* obedecem uma ordem total, pois definem a ordem em que as atividades foram concluídas.

A mineração de processos é bastante aplicável em processos de negócios fracamente definidos, onde uma definição exata do que deve ser executado não é conhecido antes da execução. Como exemplo, na área médica a conduta de dois pacientes com a mesma enfermidade pode ser diferente. Ou seja, não há uma definição completa do modelo, pois não se sabe exatamente o que deve ser executado no paciente, tal que, procedimentos excepcionais podem ser necessários, comum em atendimentos de emergência. Assim, a mineração de processos torna-se uma ferramenta essencial para que tais processos ou condutas médicas sejam conhecidas.

Em processos de negócios fracamente definidos, a execução de algumas atividades comuns pode ser violada, ou a execução de uma atividade “incomum” pode ser necessária. Desta forma, tais processos são vulneráveis a execuções fraudulentas, fator motivador para desenvolvermos técnicas para detecção desses eventos *anormais* nos *logs*. Assim, estamos interessados em descobrir quais instâncias de processos de negócios, controladas por tais sistemas gerenciadores de processos, sugerem ser uma execução anormal, anômala.

O estágio atual de desenvolvimento de pesquisas na área de mineração de processos é dedicado principalmente no apoio à definição de processos de negócios, ou seja, como melhorar os processos de negócios existentes. Neste projeto, aplicaremos a mineração de processos como ferramenta auxiliar na seleção de *traces* anômalos. No estágio atual de desenvolvimento do projeto, estamos utilizando o algoritmo de mineração de workflow descrito em [6].

Método de seleção de *traces* anômalos

O algoritmo que será apresentado a seguir corresponde a uma primeira proposta para o problema de identificar em um *log* quais ocorrências caracterizam uma anomalia. Em [5], duas propostas de aplicação da mineração de processos em segurança são apresentadas. Entretanto, tais propostas consideram a existência de dois *logs*, um para ser auditado, e outro para construir um modelo considerado padrão, formado a partir de *traces* “normais”. Consideramos essa premissa um fator limitante da proposta apresentada, mas ao mesmo tempo motivador a futuras pesquisas, pois carece de uma investigação maior na descoberta de métodos de seleção de *traces*

anômalos.

Portanto, estamos interessados em observar completamente um único *log*, e então selecionar aquelas ocorrências que caracterizam uma execução anômala. Neste projeto assumimos que um *trace* anômalo é um evento raro e não é uma instância de um modelo construído a partir de *traces* “normais”. Para tanto, adotamos o seguinte raciocínio: os *traces* que não são anomalias, quando minerados gerarão um modelo de workflow “normal”, e tentar unir um *trace* anômalo nesse modelo possivelmente requerirá mudanças estruturais. Assim, um *trace* anômalo é um *trace* que necessita modificar um modelo gerado a partir de outros *traces* do *log*.

O algoritmo proposto é como segue:

1. Seja \mathbf{L} o *log* com todos os *traces*;
2. Seja \mathbf{D} o conjunto de todos os diferentes *traces* de \mathbf{L} , onde $\mathbf{D} \subseteq \mathbf{L}$;
3. Para cada $\mathbf{t} \in \mathbf{D}$;
 - (a) $\mathbf{L}' = \mathbf{L} - \{\mathbf{t}\}$;
 - (b) Seja \mathbf{R} o conjunto de 50% dos elementos de \mathbf{L}' , aleatoriamente escolhidos;
 - (c) Seja \mathbf{M} o modelo gerado a partir de \mathbf{R} ;
 - (d) Se \mathbf{t} não for instância de \mathbf{M} , então marcar \mathbf{t} como anômalo;

Metodologia e experimentos

A imprecisão na definição de *trace* anômalo, motivou-nos a gerar exemplos de teste que caracterizariam ocorrências anômalas. Para tanto, geramos todos os *traces* possíveis, a partir de um modelo previamente concebido. Tal conjunto de *traces* representaria as ocorrências normais, enquanto os *traces* que se desviassem desse padrão representariam as ocorrências anômalas. Podemos citar como exemplo de ruído, a supressão de uma atividade executada concorrentemente no modelo original (bloco *AND*), ou a execução paralela de uma atividade executada de forma alternativa (bloco *OR*).

Para avaliarmos o algoritmo, geramos três grupos de teste, organizados de acordo com a composição dos *logs*: (i) apenas com os *traces* normais; (ii) com os *traces* normais mais alguns *traces* anômalos; e (iii) com os *traces* normais mais alguns *traces* anômalos duplicados. Os *traces* anômalos adicionados correspondiam aproximadamente 5% dos *traces*. Então, aplicamos os testes sobre os três grupos, repetindo o experimento três vezes. Consulte [7] para uma descrição mais detalhada da metodologia e dos experimentos realizados. Após o experimento, observamos que nenhum *trace* normal foi selecionado como anômalo. A seguir, detalhamos os resultados obtidos, organizados por grupo.

Primeiro grupo Nenhum *trace* selecionado como anômalo.

Segundo grupo Todas as ocorrências de *traces* anômalos selecionadas.

Terceiro grupo Alguns *traces* anômalos não foram selecionados (falsos negativos).

Referências

- [1] Wil M. P. van der Aalst, Ton Weijters, and Laura Maruster. Workflow mining: Discovering process models from event logs. *IEEE Trans. Knowl. Data Eng.*, 16(9):1128–1142, 2004.
- [2] A. de Medeiros, W. van der Aalst, and A. Weijters. Workflow mining: Current status and future directions. In Z. Tari R. Meersman and D.C. Schmidt, editors, *On The Move to Meaningful Internet Systems*, volume 2888 of *LNCS*, 2003.
- [3] L. Maruster, W. M. P. van der Aalst, T. Weijters, A. van den Bosch, and W. Daelemans. Automated discovery of workflow models from hospital data. In Kröse, B., Rijke, M., Schreiber, G., and Someren, M., editors, *Proceedings of the 13th Belgium-Netherlands Conference on Artificial Intelligence (BNAIC 2001)*, pages 183–190, 2001.
- [4] Markus Hammori, Joachim Herbst, and Niko Kleiner. Interactive workflow mining - requirements, concepts and implementation. *Data Knowl. Eng.*, 56(1):41–63, 2006.
- [5] Wil M. P. van der Aalst and Ana Karla A. de Medeiros. Process mining and security: Detecting anomalous process executions and checking process conformance. *Electr. Notes Theor. Comput. Sci.*, 121:3–21, 2005.
- [6] Jacques Wainer, Kwanghoon Kim, and Clarence A. Ellis. A Workflow Mining Method Through Model Rewriting. In Hugo Fuks, Stephan Lukosch, and Ana Carolina Salgado, editors, *Groupware: Design, Implementation, and Use: 11th International Workshop*, volume 3706, pages p. 184–19, Porto de Galinhas, Brazil, Setembro 2005. CRIWG 2005.
- [7] F. Bezerra and J. Wainer. Selecting anomalous traces from workflow event logs. SBSC 2006. Natal, RN. Novembro, 2006.

12. Workflow-based Sensor Data Management

Authors: Gilberto Zonta Pastorello Jr. and Claudia Bauzer Medeiros

There has been an increase in sensor technology to monitor all kinds of phenomena from satellite imagery to intelligent homes, from high above the Earth to deep below its surface. Examples include monitoring of water quality, animal migration tracking, traffic control, biodiversity and even biosecurity. The devices have evolved in several directions, ranging from big satellites that can produce images used for crop prediction to tiny movement, light and sound sensors, which can be spread over a house, organized as a network to control the environment aiming at safety, security and comfort. This is causing the so-called “data deluge”, posing several challenges in, for instance, data summarization, data cleaning, selective storage and data streams processing.

Using small sensors with communication capabilities is currently an important research challenge [1–3, 5, 7, 11, 12], giving birth to the sensor networks [6, 13] area. Several issues have still to be tackled in this area, including efficient energy consumption, communication protocols, data processing and transmission, self-regulated organization, and others.

The main goal of this thesis is to propose an infrastructure to support sensor data management, mainly sensor networks data, aiming at data publication and usage at large. The idea is to combine Workflow Management Systems (WFMSs) [14], Digital Content Components [8–10] and Semantic Web concepts [4] in the management of the produced data.

We plan on using a specific kind of components, called DCCs, to encapsulate the sensing devices and develop management techniques for these devices. As will be seen, DCCs offer homogeneous interfaces and encapsulate data and executable code using a single mechanism. Thus, workflows can be specified to manipulate these components, managing sensor data production via DCC interfaces. Another important issue that is going to be explored is the automatic semantic annotation of sensor data, enriching search possibilities. Semantic annotations will be considered not only to be applied to data, but to operations as well. Again, this will take advantage of the DCC model.

This project will be centered on the data production management and publication aspects of sensing devices. Issues such as fault tolerance or data transmission within a network of sensors, will not be considered. As made clear in [6], sensor data cannot be dealt with as regular heterogeneous data. Thus their management requires novel approaches.

The main advantages of our approach are:

- (i) *Encapsulating sensors in components*: this fosters interoperability, provides access homogeneity and makes composition easier. These characteristics are important when publishing the data for applications and for building new management solutions;

- (ii) *Managing components via workflows*: provides a user-centered composition solution, which facilitates construction of compositions and also their execution in a homogeneous environment;
- (iii) *Semantically annotating sensor data*: data and operation annotation provides richer documentation, enhancing not only data publication but also the creation of new management solutions. Data traceability is also an important side effect.

The challenges yet to be faced include:

DCC: (i) which characteristics from the DCC model are important, necessary or need extensions; (ii) how to specify new DCCs in terms of taxonomic referencing, interface specification and behavior implementation; (iii) how to deal with the produced data within the DCC model, considering data streams, data storage, data annotation and data validation.

Workflows: (i) how to specify workflows to manage DCCs, including user interface aspects and validation of the specifications; (ii) how to deal with workflow execution, considering automated workflow triggering, developing follow-through techniques for the user, and dynamic component association during an execution; (iii) how to enrich summarization techniques combining parts of solutions; (iv) fault verification and exception handling in an execution.

Annotation: (i) which annotation mechanisms should be used, mainly with respect to automatically annotating data; (ii) how to determine what can be annotated and how to annotate it and publish these annotations, taking into account the granularity of the annotations; (iii) how to use the annotations, in queries or associated with data streams, for instance;

Acknowledgements

This work is supported by FAPESP (grant no. 04/14052-3) and partially financed by CNPq WebMaps and AgroFlow projects.

Referências

- [1] ACM Conference On Embedded Networked Sensor Systems. 2003 and 2004 editions.
- [2] ACM Int. Conference on Mobile Computing and Networking. 1999 to 2004 editions.
- [3] ACM Int. Conference on Wireless Sensor Networks and Applications. 2002 and 2003 editions.

- [4] T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. *Scientific American*, 284(5):35–43, 2001.
- [5] Com. of the ACM. June 2004 edition – Special Edition on “Wireless sensor networks”.
- [6] J. M. Hellerstein, W. Hong, and S. Madden. The Sensor Spectrum: Technology, Trends and Requirements. *SIGMOD Record*, 32(4):22–27, 2003.
- [7] IEEE Computer. August 2004 edition – Special Edition on “Sensor Networks”.
- [8] A. Santanchè and C. B. Medeiros. Geographic Digital Content Components. In *Proceedings of VI Geoinfo*, 2004.
- [9] A. Santanchè and C. B. Medeiros. Managing Dynamic Repositories for Digital Content Components. In *Proc. of IX International Conference on Extending Database Technology – EDBT 2004 PhD Workshop*, 2004.
- [10] A. Santanchè and C. B. Medeiros. Self Describing Components: Searching for Digital Artifacts on the Web. In *Proceedings of XX SBBD*, 2005.
- [11] SIGMOD Record. December 2003 edition – Special Edition on “Sensor Network Technology and Sensor Data Management”.
- [12] SIGMOD Record. March 2004 edition – Special Edition on “Sensor Network Technology and Sensor Data Management (Part II)”.
- [13] R. Szewczyk, E. Osterweil, J. Polastre, M. Hamilton, A. Mainwaring, and D. Estrin. Habitat Monitoring with Sensor Networks. *Com. of the ACM*, 47(6):34–40, 2004.
- [14] Wikipedia. Wikipedia – The Free Encyclopedia – Workflow. <http://en.wikipedia.org/wiki/Workflow> (as of April 2006).

13. Composicionalidade de Escalonamento de Processos em Sistemas de Workflow

Autores: Gregório Baggio Tramontina e Jacques Wainer

Os sistemas de workflow atuais alocam os itens de trabalho (casos) para seus participantes baseados em dois critérios principais: a ordem de chegada dos casos na fila de um participante ou a própria iniciativa deste último em executar uma dos casos disponíveis para ele. Enquanto no primeiro caso o sistema faz a escolha da próxima instância a ser atendida, no segundo o próprio participante realiza a escolha, baseada em critérios pessoais, dentre todas as instâncias disponíveis a ele. Mas em sistemas com cargas de trabalho pesadas, essas políticas podem não ser as mais adequadas, como mostram [5–7]. O problema então se torna descobrir se a aplicação de técnicas da área de escalonamento nos sistemas de workflow, para (re)ordenar a execução dos casos nestes sistemas, produz resultados melhores que as políticas padrão mencionadas, e como essa aplicação pode ser feita de maneira a reter os benefícios observados nesses resultados.

Muitos autores reconhecem aspectos de escalonamento em workflow, como [1–4, 8, 9]. Ainda, segundo esta mesma literatura, genericamente existem quatro estruturas básicas de construção de um processo de workflow: (i) o roteamento seqüencial, mais simples, ocorre quando duas atividades devem ser executadas em seqüência; (ii) o roteamento paralelo (*AND*), quando há duas ou mais atividades que podem ser executadas simultaneamente; (iii) o roteamento condicional ou seletivo (*OR*), que acontece quando a próxima (ou próximas) atividade a ser executada depende de uma escolha, que por sua vez depende das propriedades específicas de cada caso; e (iv) o roteamento iterativo, ou *loop*, que representa atividades que devem ser executadas e, se necessário, reexecutadas, até que determinada circunstância seja atingida;

Com estes fatos em vista, o projeto concentra-se nos seguintes objetivos principais:

- estudar as construções básicas já mencionadas, simulando a utilização de técnicas de escalonamento em cada uma delas, com cargas de trabalho variadas, para se descobrir qual(is) técnica(s) traz(em) os melhores resultados para cada construção em relação às métricas-objetivo desejadas.
- estudar a possibilidade de, utilizando o conhecimento adquirido na etapa anterior, desenvolver maneiras de compor processos de workflow maiores e mais complexos com as construções básicas, de maneira a se beneficiar dos resultados já conhecidos para estas últimas; a essa etapa dá-se o nome de *composicionalidade*.

Serão feitas simulações nos cenários de workflow estudados, e que envolvem técnicas de escalonamento como *dispatching rules* e algoritmos genéticos (GAs). As *dispatching rules* são regras de atuação local, que guiam a escolha do próximo caso a

ser executado em uma atividade; assim que a atividade se torna livre para execução, o caso em sua fila que melhor satisfaz a regra é processado. As selecionadas para este projeto são FIFO (First In First Out), SIRO (Service In Random Order), EDD (Earliest Due Date) e SPT (Shortest Processing Time). Os GAs são métodos que “enxergam” o problema de escalonamento como um todo, e não apenas os jobs que estão na fila de uma máquina específica. As técnicas serão avaliadas em relação a três métricas: a *porcentagem média de casos atrasados*, a *porcentagem média de atraso*, e o *tempo médio de processamento* dos casos no sistema.

Para as simulações, foram definidos 17 cenários de workflow que contemplam as quatro estruturas básicas de roteamento mencionadas anteriormente, e que possuem três características fundamentais: (i) são dinâmicos, isto é, enquanto há casos novos chegando no sistema, outros ainda estão em processamento; (ii) existem incertezas quanto ao tempo de processamento $p_{i,j}$ de um caso j em uma atividade i , o que significa que uma entidade externa (que não seja o processo de simulação em si) não conhece o exato valor de $p_{i,j}$; (iii) em um *OR*, também existem incertezas quanto à rota que o caso vai seguir dentro desse *OR*. Para modelar essas incertezas, propõe-se aqui a metodologia *guess and solve*, que consiste em realizar uma previsão controlada sobre os tempos de processamento e rotas dos casos (a fase de *guess*) e depois usar esta previsão como argumento para as técnicas de escalonamento adequadas (a fase de *solve*).

O resultados das simulações levarão a um conjunto de “mapas” que indicam, para cada combinação de parâmetros em cada cenário estudado, as melhores técnicas de escalonamento em relação às métricas analisadas. A partir destes mapas, pode-se realizar a composicionalidade da seguinte forma: criam-se cenários compostos de duplas das estruturas básicas, por exemplo, um *OR* cujos caminhos alternativos sejam seqüências; depois, busca-se nos mapas quais as melhores técnicas para essas duas estruturas, e aplica-se estas melhores técnicas nas partes correspondentes do cenário composto. Faz-se a mesma coisa com as piores técnicas. Espera-se que o cenário composto usando as melhores métricas tenha um desempenho melhor do que o que usa as piores, mostrando que a composicionalidade existe e pode ser guiada pelos mapas gerados na etapa anterior.

Autalmente, resultados já foram obtidos em relação ao desempenho das técnicas de escalonamento nas quatro estruturas de roteamento básicas. Eles mostram, de maneira geral, que sempre existe uma melhor opção em relação às políticas tradicionais de atribuição de tarefas em workflow, seja ela uma *dispatching rule* ou o GA. Cada cenário apresenta sua peculiaridade, mas todos possuem seu conjunto de melhores técnicas. A montagem dos mapas deve começar logo mais e ficar pronta antes do fim de 2006, marcando o fim da primeira grande fase do projeto. Espera-se que a fase de análise da composicionalidade esteja pronta até meados de 2007, e que a defesa da tese se dê nesta época.

O objetivo final da pesquisa é fornecer subsídios para que os administradores dos sistemas de workflow possam decidir quando e como utilizar escalonamento, tendo material para pesar os prós e contras de tal uso em seus processos. Também se espera

contribuir com a comunidade de workflow em geral, mostrando aos desenvolvedores dessa tecnologia maneiras de se obter melhores resultados de seus sistemas, para que melhores ferramentas estejam disponíveis no mercado no médio e longo prazo.

Referências

- [1] C. Bettini, X. S. Wang, and S. Jajodia. Temporal reasoning in workflow systems. *Distributed and Parallel Databases*, 11:269–306, 2002.
- [2] J. Eder, E. Panagos, H. Pozewaunig, and M. Rabinovich. Time management in workflow systems. In *Proceedings of the BIS'99 3rd International Conference on Business Information Systems*, pages 286–300. Springer-Verlag, 1999.
- [3] Manuel Laguna and J. Marklund. *Business Process Modeling, Simulation and Design*. Prentice Hall, 2004.
- [4] P. Senkul, M. Kifer, and I. H. Toroslu. A logical framework for scheduling workflows under resource allocation constraints. In *Proceedings of the 28th International Conference on Very Large Data Bases*, pages 694–705, 2002.
- [5] Gregório Baggio Tramontina and Jacques Wainer. Modeling the behavior of dispatching rules in workflow systems: a statistical approach. In *Proceedings of the 11th International Workshop on Groupware (CRIWG) 2005*, Lecture Notes In Computer Science (LNCS), pages 208–215. Springer-Verlag Berlin Heidelberg, 2005.
- [6] Gregório Baggio Tramontina and Jacques Wainer. Statistical analysis of the performance of scheduling techniques in sequences of workflow activities. In *Anais do II Workshop Brasileiro de Tecnologias para Colaboração (WCSCW) 2005*, 2005.
- [7] Gregório Baggio Tramontina, Jacques Wainer, and Clarence Ellis. Applying scheduling techniques to minimize the number of late jobs in workflow systems. In *Proceedings of the 2004 ACM Symposium on Applied Computing*, pages 1396–1403, New York, NY, USA, 2004. ACM Press.
- [8] Wil van der Aalst and Kees van Hee. *Workflow Management: Models, Methods and Systems*. The MIT Press - Massachusetts Institute of Technology, 2002.
- [9] J. Leon Zhao and Edward A. Stohr. Temporal workflow management in a claim handling system. In *Proceedings of the international joint conference on Work activities coordination and collaboration*, pages 187–195, New York, NY, USA, 1999. ACM Press.

14. Software Architectures for Distributed Main-Memory Databases

Authors: Gustavo Maciel Dias Vieira, Anderson Supriano, and Luiz Eduardo Buzato

In recent years, web applications have become common place in most on-line businesses. These systems require fast interactive response times and serve a large number of clients. At the core of a web application lies a database system, responsible for storing its state. For many applications, as systems grow in number of customers, studies have shown that the database becomes a bottleneck [1]. Simply put, this happens because it is usually much simpler to replicate the other web service components, i.e. web servers and application servers, as they are often stateless. Commercial solutions to database replication are more concerned with high availability than with increasing the throughput, mostly because of the poor performance found in the first replication architectures [4].

However, database replication theory has advanced in the last years, promising scalable replication solutions [5]. These solutions favor the eager replication approach, maintaining database consistency and simplifying database access semantics. At the same time, commodity hardware began to be arranged in very powerful clusters of workstations [6]. These clusters are assembled from readily available parts and powered by free software, proving to be a fertile ground for the appearance of high performance parallel and replicated applications. In addition, many commodity systems now sport RAM capacities that rival that of hard disks from 5 years ago.

One way to explore this relative increase in main memory capacity is the use of main-memory databases (MMDBs) [3], capable of storing all data in main memory, thus simplifying many of the common database operations (concurrency control, locking, buffer management, recovery, etc.) and leading to a more efficient transaction processing engine. In a clustered environment, MMDBs can be replicated or distributed in the same way as disk-based databases, providing gains in terms of fault tolerance and, if the new replication algorithms are used, good scalability.

To our knowledge, there aren't commercial systems or research prototypes combining the promising approaches of eager replication and main-memory databases. In this research, we are going to develop and assess alternative software architectures and algorithms to support main-memory database replication in high performance commodity clusters. Specifically, we are going to select an existing main-memory database management system and integrate instances of this system through a replication layer.

The first step in this process is to assess the performance and general reliability of main-memory based SGBDs. We selected the Monet [2] database as the MMDB for this first evaluation. The Monet system was designed to extract maximum database performance from modern hardware, especially for complex queries. It focused on

efficient use of registers and cache memory through data structures optimized for main memory, vertical fragmentation and vectorized query execution. Monet is still being actively developed as an open source project.

To quantitatively assess Monet we used an implementation of the TPC-W benchmark created in the University of Wisconsin-Madison. This benchmark was defined by the Transaction Processing Performance Council (www.tpc.org) and provides a standard metric (WIPS - web interactions per second) for the performance of web applications. A web application is the combination of web pages, business logic and data implementing a service. In TPC-W, the web application is a book store; the benchmark defines the complete structure of the application (web pages layout, data schema, etc.), and it quantifies three load profiles implemented by simulated web browsers. Each load profile (browsing, ordering and shopping) is defined in terms of the average percentage of write operations contained in the load. Using this benchmark we compared Monet to two very commonly used open-source databases: MySQL and PostgreSQL.

The data obtained in this experiment indicate that the performance of Monet is generally the same of the others databases analyzed. PostgreSQL was the general best performer and Monet and MySQL contended for the second place. This was a bit surprising, as we were expecting Monet to perform, at least, two times faster than its rivals. Moreover, during the tests, we discovered that the stability of the SQL front-end of Monet was very weak and that this front-end still doesn't implement many useful features, like transaction isolation. After these results, we decided Monet isn't a very suitable candidate for replication.

Referências

- [1] C. Amza, A. Chanda, E. Cecchet, A. Cox, S. Elnikety, R. Gil, J. Marguerite, K. Rajamani, and W. Zwaenepoel. Specification and implementation of dynamic web site benchmarks. In *Fifth Annual IEEE International Workshop on Workload Characterization (WWC-5)*, pages 3–13, November 2002.
- [2] P. A. Boncz and M. L. Kersten. Monet: An Impressionist Sketch of an Advanced Database System. In *Proceedings Basque International Workshop on Information Technology*, San Sebastian, Spain, July 1995.
- [3] Hector Garcia-Molina and Kenneth Salem. Main memory database systems: An overview. *IEEE Transactions on Knowledge and Data Engineering*, 4(6), December 1992.
- [4] Jim Gray, Pat Helland, Patrick O'Neil, and Dennis Shasha. The dangers of replication and a solution. In *SIGMOD '96: Proceedings of the 1996 ACM SIGMOD international conference on Management of data*, pages 173–182, New York, NY, USA, 1996. ACM Press.

- [5] Bettina Kemme and Gustavo Alonso. A new approach to developing and implementing eager database replication protocols. *ACM Trans. Database Syst.*, 25(3):333–379, 2000.
- [6] T. Sterling, D. Savarese, D. J. Becker, J. E. Dorband, U. A. Ranawake, and C. V. Packer. BEOWULF: A parallel workstation for scientific computation. In *Proceedings of the 24th International Conference on Parallel Processing*, pages I:11–14, 1995.

15. WiMAX - Acesso à Internet com Qualidade de Serviço

Autores: Juliana Freitag Borin e Nelson L. S. da Fonseca

O padrão IEEE 802.16 [4], freqüentemente referenciado como WiMAX (*Worldwide Interoperability for Microwave Access Forum*), vem sendo desenvolvido com a finalidade de padronizar a tecnologia de acesso sem fio em banda larga. O padrão define a interface aérea e o protocolo de acesso ao meio para redes metropolitanas sem fio e permite altas taxas de transmissão para o acesso à Internet com baixo custo.

A arquitetura de uma rede que utiliza o padrão 802.16 possui dois elementos principais: *Base Station* (BS) e *Subscriber Station* (SS). A BS realiza a comunicação entre a rede sem fio e a rede núcleo e suporta interfaces IP, ATM, Ethernet e E1/T1. A SS fornece ao usuário acesso à rede núcleo através do estabelecimento de conexões com a BS em uma topologia Ponto-Multiponto (PMP).

A camada física opera em um formato de frames, que são subdivididos em intervalos de tempo chamados *slots* físicos. Em cada frame há um subframe *downlink* e um subframe *uplink*. O subframe *downlink* é utilizado pela BS para a transmissão de dados e de informações de controle para as SSs. O subframe *uplink* é compartilhado entre todas as SSs para transmissões com destino na BS.

Para dar suporte à grande diversidade de aplicações disponíveis na Internet, tais como serviços de voz, vídeo, multimídia e transferência de arquivos, o padrão define mecanismos de sinalização entre a BS e as SSs e quatro níveis de serviço: UGS (*Unsolicited Grant Service*), rtPS (*Real-Time Polling Service*), nrtPS (*Non-real-time Polling Service*) e BE (*Best Effort*). Em ambas as direções, *uplink* e *downlink*, os pacotes são associados à um fluxo de serviço pela camada de controle de acesso ao meio. Um conjunto de parâmetros de QoS, dentre eles retardo máximo e largura de banda mínima, é associado a cada fluxo.

Embora inclua funcionalidades para a provisão de QoS, o padrão IEEE 802.16 não sugere quais disciplinas de escalonamento devem ser utilizadas. O objetivo de um mecanismo de escalonamento é garantir a banda passante requisitada pelas SSs, bem como promover o uso eficiente do enlace sem fio. Dessa forma, esse mecanismo é fundamental para a provisão de QoS e tem grande influência no desempenho da rede. Outro mecanismo importante para a provisão de QoS não definido no padrão é o controle de admissão, responsável por limitar o número de fluxos simultaneamente ativos na rede evitando o saturamento do canal sem fio e, conseqüentemente, o não cumprimento do contrato de serviço.

Assim sendo, o objetivo desse projeto é investigar mecanismos de QoS para redes 802.16 de forma que os requisitos das diferentes aplicações, como retardo, vazão e taxa de perda de pacotes, sejam atendidos.

A metodologia escolhida para avaliar o desempenho dos mecanismos que serão propostos é a simulação de eventos discretos utilizando a ferramenta *Network Simulator* (ns). O ns é uma ferramenta de domínio público amplamente utilizada pela

comunidade científica. No entanto, devido ao padrão 802.16 ter sido recentemente aprovado, inexistente um módulo para simulação dessas redes no simulador.

Essa deficiência motivou a implementação de um módulo para simulação da camada MAC do padrão 802.16. A implementação baseou-se em um módulo [3] desenvolvido para o *ns* para simulação do padrão DOCSIS [2]. A reutilização deste código foi possível pois a especificação da camada MAC do padrão IEEE 802.16 é baseada no padrão DOCSIS [1]. Assim como no padrão 802.16, no padrão DOCSIS existe uma entidade central que controla o acesso ao meio fazendo a alocação de banda para as estações cliente. A alocação da banda baseia-se nas requisições enviadas pelas estações clientes e nos requisitos de QoS das conexões.

O módulo DOCSIS implementa três níveis de serviço para o tráfego *uplink*: UGS, rtPS e BE. Para o tráfego *downlink*, o módulo não fornece diferenciação de serviços. Assim sendo, foi necessário implementar o serviço nrtPS para o tráfego *uplink* e os quatro tipos de serviço para o tráfego *downlink*.

O DOCSIS é um padrão utilizado para transmissão de vídeo em redes cabeadas, portanto, foi preciso modificar a interface entre a camada MAC e a camada física para que o módulo 802.16 utilizasse o meio sem fio disponível no *ns*.

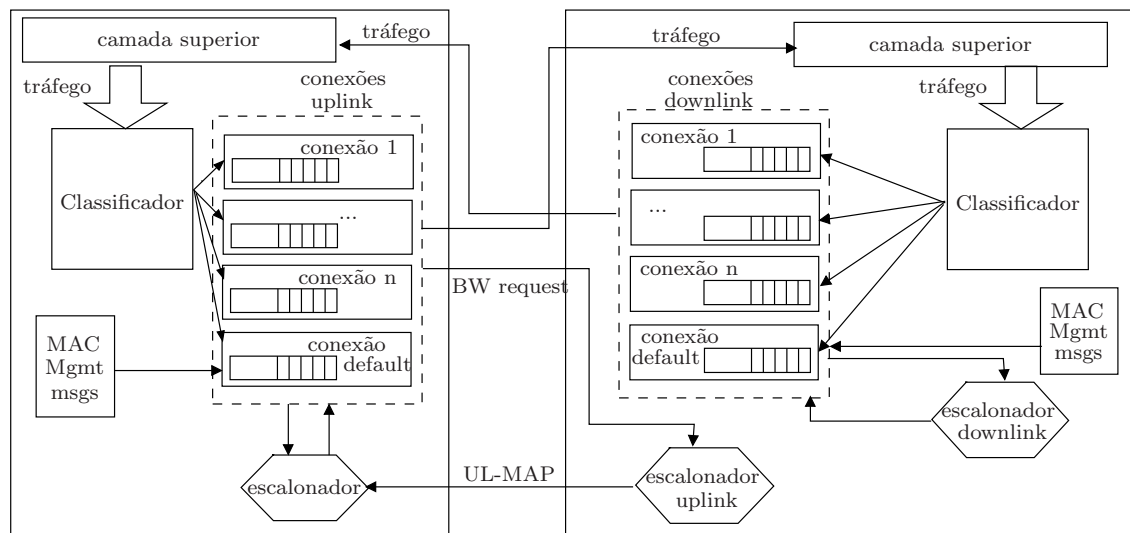


Figura 2: Estrutura do módulo para simulação do padrão IEEE 802.16

A figura 2 apresenta a estrutura do módulo implementado. Tanto na BS quanto nas SSs o tráfego proveniente da camada superior é classificado para a conexão apropriada a partir do tipo do tráfego e do par origem e destino. As conexões são configuradas pelo usuário, que entre outras informações, determina os requisitos de QoS. Cada nó da rede deve ter uma conexão *default* para transmitir mensagens de gerenciamento e o tráfego que não puder ser mapeado para nenhuma das outras conexões.

A BS possui um escalonador *uplink* e um escalonador *downlink*. O escalonador *downlink* determina quais pacotes enviados pela camada superior serão transmitidos

em cada subframe *downlink*. Para tomar essa decisão, o escalonador baseia-se nas requisições de QoS e na situação da fila de cada conexão. O escalonador *uplink* decide quais SSs terão direito de transmitir em cada subframe *uplink* e quantos *slots* cada uma dessas SSs poderá utilizar. A decisão desse escalonador leva em consideração os requisitos de QoS das conexões *uplink*, bem como as requisições de banda por elas enviadas.

No padrão 802.16, no início de cada subframe *downlink*, a BS envia um mapa *downlink* (DL-MAP) e um mapa *uplink* (UL-MAP) para informar as SSs quais delas receberão pacotes no subframe *downlink*, quais poderão transmitir no subframe *uplink* e o número de *slots* reservados para cada transmissão. No módulo implementado o DL-MAP não é enviado, dado que as informações que ele conteria não são necessárias em um sistema de simulação. Porém, no início do subframe *downlink*, são reservados alguns *slots* para simular a sobrecarga do envio dessa mensagem. O UL-MAP é enviado e contém a descrição dos *slots* alocados no subframe *uplink* subsequente.

Nas SSs existe um escalonador que decide quais pacotes provenientes da camada superior serão enviados nos *slots* recebidos. Para tomar essa decisão o escalonador baseia-se nas informações trazidas pelo UL-MAP, nos requisitos de QoS e na situação das filas das conexões.

O presente módulo passou por um processo de validação e os resultados indicam que seu funcionamento está de acordo com o padrão 802.16. Porém, por falta de espaço, os gráficos com os resultados não são apresentados.

Referências

- [1] C. Eklund *et al*, IEEE Standard 802.16: A Technical Overview of the WirelessMAN Air Interface for Broadband Wireless Access. *IEEE Communications Magazine*, vol. 40, no. 6, pg. 98-107, 2002.
- [2] Cable Television Labs Inc., Cable Labs, *Data Over Cable Service Interface Specifications - Radio Frequency Interface Specification*. SP-RFIV2.0.
- [3] *DOCSIS Research Project*. <http://www.cs.clemson.edu/~jmarty/docsis.html>.
- [4] *IEEE Standard for Local and Metropolitan Area Networks - Part 16: Air Interface for Fixed Broadband Wireless Access Systems*. IEEE Std., Rev. IEEE Std802.16-2004, 2004.
- [5] G. Nair *et al*, IEEE 802.16 Medium Access Control and Service Provisioning. *Intel Technology Journal*, vol. 8, no. 3, pg. 213-228, 2004.
- [6] *The Network Simulator - ns-2*. <http://www.isi.edu/nsnam/ns/>, 2002.

16. Análise/Otimização de Potência de SoCs Heterogêneos Ponderando

Autores: Felipe Klein e Rodolfo Azevedo

Com a constante redução no tamanho dos transistores e o conseqüente aumento no número de transistores por *chip*, a potência dissipada pelos circuitos digitais está crescendo exponencialmente. Aliado a isto, a demanda por dispositivos portáteis, tais como PDAs, MP3 *players*, telefones celulares, entre outros, não pára de crescer e, a cada nova geração destes produtos, mais funcionalidades são agregadas que ocasionam o aumento de sua complexidade e conseqüentemente do consumo de energia.

Este aumento de complexidade foi previsto em meados da década de 70, na famosa “Lei de Moore”, que estabeleceu que o número de transistores dobraria a aproximadamente cada 18 meses [1]. Uma das implicações do aumento da dissipação de potência é a redução do tempo de vida útil do circuito, pois o aumento da densidade de potência ($\frac{W}{cm^2}$) pode levar o circuito a falência.

Outro resultado do crescente aumento da dissipação de potência é o inevitável aumento do custo final do produto, devido a elaboração de soluções sofisticadas para o resfriamento do *chip*. Este é um fator importante, que pode tornar o produto inviável do ponto de vista do mercado. Uma outra implicação importante, principalmente para os dispositivos portáteis, geralmente alimentados por bateria, é que o aumento do consumo de energia pode reduzir drasticamente a autonomia do dispositivo, podendo assim inviabilizar o seu uso na prática.

Por isso, fica claro que o projeto de sistemas digitais visando a redução do consumo de energia se torna um elemento-chave no fluxo de projeto – o chamado *low power design*.

Porém, ao contrário de ferramentas para análise de desempenho e área, que têm um grande leque de opções disponíveis, tanto na indústria quanto no meio acadêmico, relativamente poucas opções estão disponíveis para a análise/otimização de potência em níveis mais altos de abstração.

Para os níveis mais baixos, como de transistor (*transistor-level*) e de portas lógicas (*gate-level*), há diversas opções de ferramentas de CAD¹, como o SPICE [2]. No entanto, estimar potência de circuitos médios e grandes nestes níveis pode ser uma tarefa excessivamente custosa em termos de tempo. Uma outra dificuldade é que estas ferramentas se localizam nos últimos estágios do ciclo de projeto, retardando a detecção de uma escolha de projeto mal-feita. Isso geraria novas rodadas completas no ciclo de desenvolvimento, até que as restrições de projeto fossem alcançadas.

Por isso, conforme a complexidade do sistema aumenta, é necessário prover ao projetista um meio de subir nos níveis de abstração, onde estão as maiores oportunidades de otimização para potência [3]. Sabe-se que a precisão absoluta das técnicas de estimativa de potência de alto nível são menos apuradas que as de baixo nível.

¹CAD – Computer-Aided Design

Porém, mesmo uma estimativa grosseira em alto nível pode economizar muito tempo de projeto, dado que **gargalos de consumo** podem ser detectados mais cedo, acelerando o processo de otimização.

Aliado a isso, a pressão do mercado para um *time-to-market* cada vez menor tem evidenciado a tecnologia dos chamados SoCs², que permitem a integração de processadores, memórias e uma grande variedade de módulos de *hardware* e *software*, de forma a aumentar a produtividade (via reusabilidade) durante o projeto de arquiteturas voltadas para uma aplicação específica. Como consequência, cada vez mais técnicas de estimativa/otimização de potência são necessárias para uma análise apropriada de tais sistemas.

O projeto de doutorado que vem sendo desenvolvido, é uma ampliação e continuação do trabalho de mestrado [4] do mesmo autor, realizado no Instituto de Computação da UNICAMP, e concluído em abril de 2005.

Em [4] foi desenvolvida a biblioteca PowerSC, que estende a linguagem de descrição de sistemas SystemC [5]. A PowerSC permite que os módulos (IPs³) descritos em SystemC RTL⁴ sejam analisados em função de potência, de uma maneira muito transparente para o projetista.

A próxima etapa sendo desenvolvida é uma metodologia e modelos que permitam a análise de potência de SoCs, incluindo todos os seus componentes. Os SoCs são sistemas compostos por diferentes componentes, tais como processadores de uso geral, memórias (SRAM, DRAM), *caches*, IP *cores*, ASIPs⁵, barramentos, elementos de software, etc.

SoCs são sistemas complexos e heterogêneos e, como não é difícil de notar, a análise de potência precisa de tais sistemas também é complexa. A interação entre os diversos componentes dos SoCs altera a forma como a energia é consumida no sistema. Como exemplo ilustrativo, considere um sistema com 2 processadores, 1 barramento e 1 memória. A concorrência pela memória aumenta a quantidade de *stalls* nos processadores, aumentando o consumo de energia do sistema em relação ao sistema monoprocessado. Outro fator agravante no consumo de energia é o aumento no número de transações no barramento, causado pela intensificação de tráfego de dados entre os dois processadores.

Em vista disso, uma metodologia eficiente que permita ao projetista modelar os diversos componentes (através de linguagens de descrição de arquiteturas [11], por exemplo) e, durante este tempo, efetuar *trade-offs* arquiteturais com relação à potência é essencial, mas ainda inexistente.

²SoC – System-on-Chip

³IP – Intellectual Property

⁴RTL – Register-Transfer Level

⁵ASIP – Application Specific Instruction Processor

Referências

- [1] Gordon E. Moore. Cramming more components onto integrated circuits. *Electronics*, 38(8):114–117, 19 April 1965.
- [2] L. W. Nagel. Spice2: A computer program to simulate semiconductor circuits. Erl-m520, Univ. California, Berkeley, 1975.
- [3] Jan Rabaey. Low-power system design. In *EUROCHIP Course on Methods and Tools for Digital System Design*, 1995. Leuven, Netherlands.
- [4] Felipe V. Klein. Powersc: Uma extensão de systemc para a captura de atividade de transição. Master’s thesis, Instituto de Computação, UNICAMP, April 2005.
- [5] Open SystemC Initiative. *SystemC Language Reference Manual*, revision 1.0 edition, 2003. See <http://www.systemc.org>.
- [6] Paul E. Landman and Jan M. Rabaey. Activity-sensitive architectural power analysis. In *IEEE Transactions on Computer-Aided Design of Integrated Circuits*, pages 571–587. IEEE Computer Society Press, June 1996.
- [7] Vivek Tiwari, Sharad Malik, and Andrew Wolfe. Power analysis of embedded software: a first step towards software power minimization. In *ICCAD ’94: Proceedings of the 1994 IEEE/ACM international conference on Computer-aided design*, pages 384–390. IEEE Computer Society Press, 1994.
- [8] Praveen Kalla, Jörg Henkel, and Xiaobo Sharon Hu. Sea: Fast power estimation for micro-architectures. In *ASP-DAC ’03: Proceedings of the 2003 conference on Asia South Pacific design automation*, pages 600–605, 2003.
- [9] Reinaldo A. Bergamaschi and Yunjian W. Jiang. State-based power analysis for systems-on-chip. In *DAC ’03: Proceedings of the 40th conference on Design automation*, pages 638–641. ACM Press, 2003.
- [10] F. Klein, R. Azevedo, and G. Araujo. High-level switching activity prediction through sampled monitored simulation. In *Proc. of International Symposium on System-on-Chip (SOC)*, Tampere, Finland, November 2005. Accepted for publication.
- [11] Marcus Bartholomeu Sandro Rigo, Guido Araujo and Rodolfo Azevedo. Archc: A systemc-based architecture description language. In *to appear in the 16th Symposium on Computer Architecture and High Performance Computing - Foz do Iguacu, Brazil*, October 2004.
- [12] Nikil Dutt, Alex Nicolau, Hiroyuki Tomiyama, and Ashok Halambi. New directions in compiler technology for embedded systems (embedded tutorial). In *ASP-DAC ’01: Proceedings of the 2001 conference on Asia South Pacific design automation*, pages 409–414, New York, NY, USA, 2001. ACM Press.

17. Algoritmo para Problema de Conexidade em Grafos

Autores: Luís Augusto Angelotti Meira e Flávio Keidy Miyazawa

O foco deste doutorado consiste em estudar Problemas de Conexidade em Grafos. Os problemas abordados não se enquadram na classe de problemas polinomial, Classe P . Isso implica não haver, até o momento, algoritmos exatos polinomiais para eles. Sendo assim, uma série de abordagens foram consideradas na pesquisa, desde algoritmos aproximativos e heurísticas, a algoritmos exatos não polinomiais.

Os problemas são de natureza combinatória, logo esta pesquisa pode ser classificada como *Otimização Combinatória*, um ramo da *Teoria da Computação*. Os resultados foram de interesse prático ou teórico. Algoritmos rápidos, com potencial aplicação a problemas reais, constituíram um dos focos de interesse, bem como algoritmos polinomiais sem uma aplicação direta, mas que agregaram conhecimento aos problemas.

Dentro da Otimização Combinatória, este doutorado concentrou-se em *Algoritmos Aproximados*, uma área que tem se revelado crescentemente promissora. Existe uma série de congressos renomados específicos em aproximação, como RANDOM, APPROX, WAOA. Além disso, congressos mais gerais de grande repercussão mundial, como FOCS, STOC, LATIN, WEA, ISAAC, por exemplo, têm aceito muitos artigos em aproximação.

Problemas de Conexidade em Grafos são de grande interesse para a comunidade científica ao redor do mundo. Eles têm uma enumerável gama de funcionalidades em Computação Aplicada. Todo problema que possa ser descrito por meio de grafo é um potencial candidato a utilizar esses algoritmos. Além disso, existe interesse teórico de se obter aproximações de qualidade, mesmo que a complexidade seja descrita por um polinômio de alto grau.

Em vista do exposto, é de se concluir que a pesquisa desenvolvida desperta grande interesse e tem importância considerável para a Ciência da Computação atual.

Considerando o período dispendido ao doutorado, foram alcançados resultados que merecem menção: primeiramente, sobre um problema de classificação, conhecido na literatura especializada como *Uniform Labeling Problem*, foi elaborado um artigo em conjunto com Evandro C. Bracht e Flávio K. Miyazawa [2], o qual foi aceito no congresso WEA. A versão extendida foi submetida e aceita na revista JEA, *Journal of Efficient Algorithm* [3].

Também consegui, em conjunto com meu orientador, resultados interessantes relativos ao Problema do Corte Ponderado (*Ratio Cut*).

Finalmente foi dispendido algum tempo sobre o problema K -means.

Problema da Classificação Métrica Uniforme

Em um problema de classificação tradicional, deseja-se conectar cada um dos n objetos a uma das m classes. Essa classificação deve ser consistente com alguns dados observados, os quais incluem a relação entre os pares de objetos a serem classificados e a relação entre objetos e classes.

Mais especificamente, o problema da classificação pode ser definido da seguinte forma:

PROBLEMA DA CLASSIFICAÇÃO MÉTRICA UNIFORME: Seja P o conjunto de objetos, L o conjunto de classes, $w : P \times P \rightarrow \mathbb{R}^+$ uma função peso, $d : L \times L \rightarrow \mathbb{R}^+$ uma função de distância e $c : P \times L \rightarrow \mathbb{R}^+$ uma função de custo de atribuição. Uma *classificação* de P em L é uma função $\phi : P \rightarrow L$. O *custo de atribuição* de uma classificação ϕ é a soma $\sum_{i \in P} c(i, \phi(i))$ e o *custo de separação* de uma classificação ϕ é a soma $\sum_{i, j \in P} d(\phi(i), \phi(j))w(i, j)$. A função w indica a força de relacionamento entre dois objetos, e a função d indica a similaridade entre duas classes. O custo de uma classificação ϕ é a soma dos custos de conexão com os custos de separação. O Problema da Classificação Métrica (*The Metric Labeling Problem MLP*) consiste em encontrar uma classificação de objetos em classes com custo total mínimo. Definimos os tamanhos dos conjuntos P e L como n e m , respectivamente.

Problema do Corte Ponderado (*Ratio Cut*)

Em nosso trabalho sobre o Corte Ponderado (*Ratio Cut*), foi analisada uma relaxação conhecida para o problema. Esta relaxação é baseada em programação semidefinida e foi apresentada por Arora *et al.* [1]. Foi desenvolvido também um algoritmo exato, utilizando a técnica de *branch and bound* e algumas heurísticas, todas baseadas nessa relaxação. A relaxação e os algoritmos foram testados em instâncias de tamanho pequeno e moderado. A relaxação gerou valores muito próximos da solução ótima. O algoritmo exato pôde obter soluções para instâncias de tamanho pequeno e moderado e as melhores heurísticas obtiveram soluções ótimas ou quase ótimas para todas as instâncias testadas. Não obtivemos nenhum algoritmo aproximado, mas provamos características interessantes para cada uma das heurísticas.

Os algoritmos foram testados em um conjunto de instâncias aleatórias e produziram soluções ótimas ou quase ótimas. Comparamos o *branch and bound* baseado em programação semidefinida com uma formulação baseada em programação inteira [5] que foi resolvida pelo Xpress-MP [6].

Além disso, foi provada a existência de um limite de aproximação para o problema quando a esperança de ambas as partes da fração é limitada. Essa estratégia pode ser aplicada a outros problemas com uma fração na função objetivo. Cabe mencionar que, tendo em vista não existir um modo de decompor a esperança de uma fração em termos de suas partes, este resultado tem uma utilidade mais geral.

Definição do Problema

Dados um grafo $G = (V, E)$, funções $c : V \times V \rightarrow \mathbb{R}^+$ e $w : V \rightarrow \mathbb{R}^+$, e um conjunto $S \subset V$, denotamos por $\delta(S)$ o conjunto de arestas com exatamente uma extremidade em S , $\mathcal{C}(S)$ a soma $\sum_{e \in \delta(S)} c_e$, $\mathcal{W}(S)$ o produto $w(S)w(V \setminus S)$, onde $w(C) := \sum_{v \in C} w_v$, e $\rho(S)$ o valor $\mathcal{C}(S)/\mathcal{W}(S)$. O problema do corte ponderado pode ser definido da seguinte maneira:

PROBLEMA DO CORTE PONDERADO: Dados um grafo $G = (V, E)$, custos $c_e \in \mathbb{R}^+$ para cada aresta $e \in E$, e pesos $w_v \in \mathbb{R}^+$ para cada vértice $v \in V$, encontre um corte $S \subset V, S \neq \emptyset$, que minimize $\rho(S)$.

Na versão sem pesos, nós temos $c_e = 1$ se $e \in E$ e $c_e = 0$ caso contrário, e $w_v = 1$ para cada v em V .

O problema do corte ponderado também é chamado de *Sparsest Cut problem* e *Min Flux Cut problem* [4].

Referências

- [1] Sanjeev Arora, Satish Rao, and Umesh Vazirani. Expander flows, geometric embeddings, and graph partitionings. *ACM Symposium on Theory of Computing*, 2004.
- [2] Evandro C. Bracht, Luis A. A. Meira, and Flavio Keidi Miyazawa. A greedy approximation algorithm for the uniform labeling problem analyzed by a primal-dual technique. In *WEA*, volume 3059 of *Lecture Notes in Computer Science*, pages 145–158. WEA, Springer, 2004.
- [3] Evandro C. Bracht, Luis A. A. Meira, and Flavio Keidi Miyazawa. A greedy approximation algorithm for the uniform metric labeling problem analyzed by a primal-dual technique. *Journal of Efficient Algorithm*, 2006. Paper Accepted.
- [4] P. Crescenzi and V. Kann. *A compendium of NP optimization problems*, 1994.
- [5] K. Freivalds. A nondifferentiable optimization approach to ratio-cut partitioning. In *Proc. 2nd Workshop on Efficient and Experimental Algorithms*, Lectures Notes on Computer Science, LNCS 2647. Springer-Verlag, 2003.
- [6] Dash Optimization. *Xpress-MP Manual. Release 13*. 2002.

18. Design Inclusivo de Sistemas de Informação na Web

Autores: Amanda Meincke Melo, M. Cecilia C. Baranauskas

Entre tantas opções, a *Web* tem sido cada vez mais utilizada por milhões de pessoas para acesso à informação, editoração e publicação de textos, movimentações bancárias, transações comerciais, trocas de mensagens interpessoais, formação de comunidades virtuais, entretenimento e educação etc. Para interagir com aplicações na *Web* usufruindo plenamente e autonomamente de sua funcionalidade, entretanto, é fundamental que os seus usuários consigam perceber os elementos de interface e as outras informações que são apresentados em suas páginas, fazer sentido daquilo que é percebido e operar sobre os elementos de interface disponíveis. Muitas pessoas ainda são privadas do acesso e do uso de diversos serviços oferecidos na *Web*, mesmo quando têm acesso ao computador e à Internet. É o caso de pessoas com deficiência (ex. perceptual, cognitiva, motora), e de usuários que experimentam algum tipo de limitação imposta pelo ambiente físico (ex. ruído, iluminação inadequada, mobiliário em configuração diferente da convencional) ou pela infra-estrutura de que dispõem para acessar a *Web* (ex. resolução de tela pequena, navegadores textuais, ausência do *mouse*), entre outras condições.

Acessibilidade é um requisito contemporâneo à qualidade no uso de sistemas interativos. Um sistema que não oferece a mínima condição de acesso e de interação para um determinado usuário em uma situação específica de uso, nem mesmo pode ser avaliado adequadamente quanto à sua usabilidade. Segundo o W3C [13], a falta de acessibilidade ao conteúdo da *Web* impede muitos usuários de aproveitá-la plenamente. A ausência de estrutura nos *sites*, o uso abusivo de informações gráficas sem alternativas adequadas de texto ou outro tipo de comentário, desorientam o usuário, dificultando a sua navegação. Ainda, em termos de Brasil, acessibilidade também está relacionada à pluralidade de condições de nossa população e da urgência em pensarmos soluções que dêem conta de promover a cidadania mediada por tecnologias digitais [11].

É inegável a contribuição do usuário no desenvolvimento de produtos para seu uso, especialmente aplicações de *software*, e seu papel tem sido amplamente reconhecido em áreas como Interação Humano-Computador (IHC) e Engenharia de *Software* (ES). Embora existam métodos, técnicas, recomendações, tecnologias e ferramentas para apoiar o desenvolvimento e a avaliação de páginas *Web* que sejam acessíveis aos seus usuários, não encontramos na literatura processos de design de sistemas de informação na *Web* que considerem amplamente as diferenças entre as pessoas e as atividades que realizam em ambientes sociais inclusivos. Esta pesquisa propõe um modelo de processo para o design inclusivo de sistemas de informação na *Web*. Por design inclusivo entendemos um processo de design com o usuário, considerado em suas diferenças (físicas, cognitivas, sociais).

A Engenharia de *Software Web* [1] [2] explicita a necessidade de o design de

sistemas de informação na *Web* ser apoiado por processos sistemáticos e organizados, que dêem conta da complexidade envolvida na produção de sistemas de *software* para essa mídia. Diferentes disciplinas relacionadas ao desenvolvimento de sistemas de informação contribuem na proposta de um modelo inclusivo para design de sistemas de informação na *Web*: Semiótica Organizacional [3] [12], Design Participativo [6] [7], Interação Humano-Computador (IHC) [10], Engenharia *Web* [1] [2]. Enquanto alguns métodos da Semiótica Organizacional [3], contribuem à clarificação e à representação dos diferentes aspectos envolvidos no design de sistemas de informação, a abordagem do Design Participativo [7] orienta a participação de usuários em diferentes momentos do processo de design. O modelo da Engenharia de Usabilidade [8] [10] é tomado como base do processo, orientando a organização dos métodos e das técnicas de design e de avaliação de interfaces de usuário ao longo do ciclo de vida de design de sistemas interativos.

O modelo de processo que resulta desta pesquisa está organizado de acordo com o meta-modelo de processo SPEM - *Software Process Engineering Metamodel* [9], sugerido pela OMG - *Object Management Group*.

Como parte das contribuições apresentamos princípios gerais para o design inclusivo de sistemas de informação na *Web* e sugerimos técnicas para apoiar a participação de diferentes partes interessadas, inclusive pessoas com deficiência, na concepção, na proposição e na avaliação de sistemas de informação na *Web* [4] [5]. Estas contribuições estão apoiadas em um entendimento abrangente para o design de sistemas de informação e no reconhecimento do valor da participação genuína das diferentes partes interessadas no sistema de informação em desenvolvimento.

O modelo de processo delineado tem sido experimentado no design do portal do projeto "Todos Nós - Unicamp Acessível" e na reformulação do *site* da Diretoria Acadêmica da Unicamp, modificando a percepção de diferentes partes interessadas para o que significa promover a qualidade no uso em sistemas de informação na *Web* no contexto de uma sociedade inclusiva.

Referências

- [1] Y. Deshpande, S. Hansen. Web Engineering: Creating a Discipline among Disciplines. *IEEE MultiMedia*, April 2001, pp. 82-87.
- [2] A. Ginige, S. Murugesan. Guest Editor's Introduction: The Essence of Web Engineering. *IEEE MultiMedia*, April 2001, pp. 22-25.
- [3] K. Liu. *Semiotics in Information Systems Engineering*. Cambridge University Press, 2000, 218p.
- [4] A. M. Melo, M. C. C. Baranauskas. An Inclusive Approach to Cooperative Evaluation of Web User Interfaces. In *Proceedings of ICEIS 2006*, Chipre, pp. 65-70.

- [5] A. M. Melo, M. C. C. Baranauskas. Design Inclusivo de Sistemas de Informação na Web. In *Simpósio Sobre Fatores Humanos em Sistemas Computacionais*, Natal, Brasil, Novembro 2006.
- [6] A. Monk, P. Wright, J. Haber and L. Davenport. Appendix 1 - Cooperative Evaluation: A run-time guide. *Improving your human-computer interface: a practical technique*, Prentice-Hall, 1993.
- [7] M. J. Müller, J. H. Haslwanter, T. Dayton. Participatory Practices in the Software Lifecycle. Helander, M.G., Landauer, T.K., Prabhu, P.V. (eds.), *Handbook of Human-Computer Interaction*, 2nd edition, Elsevier, 1997, pp. 255-297.
- [8] J. Nielsen. The Usability Engineering Life Cycle. *Computer*, vol. 25, issue 3, IEE, 1992, 12-22.
- [9] OMG. Software Process Engineering Metamodel, Version 1.1. *Object Management Group*, 2005. Disponível em: <http://www.omg.org/technology/documents/formal/spem.html>
- [10] H. V. Rocha, M. C. C. Baranauskas. *Design e Avaliação de Interfaces Humano-Computador*. Nied, São Paulo, 2003, 244p.
- [11] SBC. *Seminário Grandes Desafios da Computação no Brasil - 2006 a 2016*. São Paulo, SP, 8 a 9 de Maio.
- [12] R. Stamper. Organisational Semiotics: Informatics without the computer? Liu, K. et al (eds.) *Information, organization, and technology: studies in organizational semiotics*, Kluwer Academic Publishers, Massachusetts, pp. 115 - 171.
- [13] W3C. Web Content Accessibility Guidelines 1.0. *World Wide Web Consortium*, 1999. Disponível em: <http://www.w3.org/TR/WCAG10/>. Acesso em: Ago. 2006.

19. Uma Arquitetura para Redes Mesh Sem Fio Auto-Gerenciáveis

Autores: Neumar Malheiros e Edmundo Madeira

As tecnologias de comunicação sem fio representam uma alternativa flexível e de baixo custo para estender a cobertura da infra-estrutura fixa de comunicação, a partir da implementação de redes de acesso sem fio com altas taxas de transmissão e suporte à mobilidade de usuários. Nesse contexto, a arquitetura de Redes *Mesh* Sem Fio (*Wireless Mesh Networks*) [1] emerge como um solução bastante atracente, como demonstra sua aplicação no projeto de Redes Comunitárias (*Community Networks*), que visam prover compartilhamento de informações, acesso à Internet e outros serviços de rede, a uma determinada comunidade de usuários.

A arquitetura de Redes *Mesh* Sem Fio é caracterizada por dois aspectos principais. Primeiro, os elementos de rede cooperam entre si formando uma rede *ad hoc* com encaminhamento distribuído (*multihop ad hoc network*) [2]. Segundo, um conjunto de nós estacionários, denominados roteadores *mesh*, realizam funções de roteamento e encaminhamento, formando um sistema de distribuição (*backbone*) sem fio, pelo qual as mensagens são transmitidas, nó a nó, até o destino final. Esses nós dedicados também podem funcionar como pontes ou elementos de acesso (*gateways*) a outras redes, inclusive à infra-estrutura fixa de comunicação. Entretanto, nada impede que os demais nós (denominados clientes *mesh*) se envolvam na tarefa de encaminhamento das mensagens. Dessa forma, a arquitetura de Redes *Mesh* Sem Fio possibilita a implementação, em larga escala, de redes de acesso sem fio com altas taxas de transmissão, oferecendo uma solução promissora para o desenvolvimento de uma Internet ubíqua.

No entanto, muitas questões ainda precisam ser resolvidas para que os benefícios dessa arquitetura possam ser explorados em todo seu potencial. Em particular, o controle e gerência de Redes *Mesh* são tarefas bastante complexas devido à natureza dinâmica e heterogênea de sua arquitetura, mobilidade e heterogeneidade dos elementos de rede e requisitos de auto-configuração, entre outros. De fato, as tradicionais abordagens de gerência centralizada, baseadas em um modelo cliente e servidor, não são viáveis neste contexto. Este fato também impõe limitações ao suporte que essas redes podem oferecer para o desenvolvimento, manutenção e integração de novas aplicações e serviços avançados.

Para lidar com esse problema, este projeto de pesquisa propõe investigar uma abordagem para gerência de Redes *Mesh* Sem Fio baseada nos princípios de um novo paradigma de sistemas computacionais, denominado *Computação Autônoma* (*Autonomic Computing*). Este paradigma consiste em sistemas computacionais que são capazes de gerenciar a si mesmos, de acordo com objetivos de alto nível [3]. Esta abordagem para o projeto de sistemas *auto-gerenciáveis* foi motivada pelas limitações impostas em função da crescente complexidade envolvida na gerência e operação dos atuais sistemas de informação e comunicação. Quando aplicado ao

projeto de sistemas e redes de comunicação este paradigma é denominado *Autonomic Communications*. Neste caso, o objetivo é prover comportamento autônomo, flexível e ciente de contexto aos sistemas de comunicação. Isso significa projetar arquiteturas de redes de comunicação *auto-gerenciáveis*, cuja operação é controlada por políticas de alto nível, definidas de acordo com objetivos administrativos. Nesse contexto, o sistema autônomo é a própria rede, então denominada *Rede Autônoma (Autonomic Network)*.

Existem diversos trabalhos de pesquisa com propostas de algoritmos e mecanismos para auto-gerência de redes *ad hoc* [4,5], assim como para categorias específicas de redes *ad hoc*, como Redes de Sensores Sem Fio [6,7]. Entretanto, essas propostas não podem ser utilizadas em Redes *Mesh* sem se considerar cuidadosamente as características particulares desta arquitetura.

O objetivo da presente proposta de pesquisa é utilizar princípios de *Autonomic Communications* na gerência e operação de Redes *Mesh* Sem Fio. Isso significa estabelecer os primeiros passos no projeto de Redes *Mesh* Autônomas (*Autonomic Wireless Mesh Networks*). O resultado esperado é o projeto de uma arquitetura distribuída para Redes *Mesh* Sem Fio com propriedades de auto-gerência. Em princípio, as *propriedades autônomas* consideradas são: auto-configuração e auto-otimização

Em linhas gerais, o projeto dessa arquitetura consiste em duas tarefas principais. Primeiro, definir uma *Entidade de Gerência Autônoma* que é incorporada a cada elemento de rede a fim de torná-lo um *Elemento Autônomo*. Segundo, definir mecanismos de comunicação e cooperação entre esses elementos de forma que a autonomia da rede seja o resultado da interação coordenada dos elementos autônomos. Dessa maneira, um dos principais desafios desta pesquisa é como assegurar a convergência de mecanismos distribuídos de auto-gerência regulados por políticas de alto-nível, a fim de que a rede como um todo possa alcançar um estado desejado a partir da cooperação entre seus elementos autônomos. Outros desafios envolvem algoritmos distribuídos para auto-configuração e auto-otimização; mecanismos de adaptação dos elementos às alterações no ambiente; gerência de políticas que governam o comportamento autônomo do sistema; e mecanismos para o compartilhamento de informações de gerência entre os elementos.

Referências

- [1] I. F. Akyildiz e X. Wang. A Survey on Wireless Mesh Networks. *Communications Magazine, IEEE*, 43(9):S23–S30, 2005.
- [2] R. Bruno; M. Conti; e E. Gregori. Mesh Networks: Commodity Multihop Ad Hoc Networks. *Communications Magazine, IEEE*, 43(3):123–131, 2005.
- [3] J. O. Kephart e D. M. Chess. The Vision of Autonomic Computing. *Computer, IEEE*, 36(1):41–50, 2003.

- [4] C. Shen; C. Srisathapornphat; e C. Jaikaeo. An Adaptive Management Architecture for Ad Hoc Networks. *Communications Magazine, IEEE*, 41(2):108–115, 2003.
- [5] A. Malatras; G. Pavlou; S. Gouveris; S. Sivavakeesar; e V. Karakoidas. Self-Configuring and Optimizing Mobile Ad Hoc Networks. Em *Autonomic Computing, ICAC 2005. Proceedings of the Second International Conference on*, p. 372–373, 2005.
- [6] L.B. Ruiz; T.R.M. Braga; F.A. Silva; H.P. Assuncao; J.M.S. Nogueira; e A.A.F. Loureiro. On the Design of a Self-Managed Wireless Sensor Network. *Communications Magazine, IEEE*, 43(8):95–102, 2005.
- [7] D. Marsh; R. Tynan; D. O’Kane; e G. M. P. O’Hare. Autonomic Wireless Sensor Networks. *Engineering Applications of Artificial Intelligence*, 17(7):741–748, 2004.

20. Replicação de Checkpoints em Ambientes Distribuídos

Autores: Raphael Marcos Menderico, Islene Calciolari Garcia

Motivação: *Checkpoints* são utilizados como solução para diversos problemas, entre eles a recuperação por retrocesso de estado. Nesse caso, consideramos que os estados são gravados em memória estável e, na presença de uma falha, esses *checkpoints* poderão ser lidos nessa mesma memória para permitir a construção de um *checkpoint* global consistente, para o qual os processos possam retroceder. [2]

Porém, em certos ambientes, o acesso à memória estável de um componente falho pode não ser imediato. Quando consideramos um ambiente que permita falhas do tipo *crash*, perdemos, além do processamento, o acesso aos demais componentes da máquina — entre eles discos e outros dispositivos de memória estável. Quando partimos para grades computacionais, essa falha apresenta-se na forma de um computador que, por algum motivo, foi desconectado da rede (por uma solicitação do usuário, por exemplo) [1].

Uma possível solução para este problema é associar a memória estável a um componente diferente daquele que realiza o processamento, de forma a tornar essas falhas eventos independentes e, portanto, com menor probabilidade de ocorrerem simultaneamente. Essa solução, no entanto, aumenta o custo do hardware necessário para executar a aplicação distribuída. Além disso, as condições necessárias para que os eventos sejam independentes de fato (separação física das máquinas, inclusive em termos elétricos, entre outros) podem acabar comprometendo a performance do sistema como um todo de uma forma crítica.

Outra solução possível é utilizar algum dispositivo de armazenamento global para separar o armazenamento do processamento em si. Nesse caso, os processos comunicam-se com o dispositivo de armazenamento através da rede. O problema desse modelo é que ele permite a existência de um único ponto de falha (o local onde os checkpoints serão armazenados). Além disso, ele ainda torna a gravação de *checkpoints* uma operação mais lenta e sobrecarrega o sistema de comunicação (já que há a necessidade de enviar a informação ao servidor e aguardar uma resposta para considerar que a gravação foi bem sucedida).

Para contornar esses problemas seria necessário que os protocolos fossem alterados de forma a tolerarem tais falhas, ou seja, seria preciso que um retrocesso não dependesse de uma informação gravada na memória estável do componente que apresentou a falha. Uma solução para esse problema seria copiar os *checkpoints* em outros processos da aplicação, permitindo que a aplicação retroceda somente utilizando as informações existentes nos processos que não falharam.

Objetivo: O objetivo desse trabalho é estudar técnicas de replicação de *checkpoints* e as garantias que podem ser obtidas a partir disso, em especial aquelas

relacionadas com retrocesso da computação no caso de falhas do tipo *crash*. A meta é projetar algoritmos que, dada uma aplicação distribuída, coordenem a replicação dos *checkpoints* através dos processos da aplicação e obtenham um *checkpoint* global consistente em casos de falhas do tipo *crash*, desde que estas não ultrapassem um certo número k de processos cuja memória estável não esteja disponível. Esses algoritmos deverão coordenar a gravação dos *checkpoints*, a replicação dos *checkpoints* entre os processos, a recuperação por retrocesso no caso de uma falha e a eliminação de *checkpoints* desnecessários.

Encontramos na literatura diversos exemplos de protocolos de *checkpointing* e de formas para se obter o retrocesso a partir deles. [3, 10] A grande maioria supõe que os *checkpoints* serão armazenados em um disco centralizado que pode ser acessado por qualquer processo em caso de falha. Existem alguns trabalhos que abordam a replicação de *checkpoints* [1, 4–6, 9], entretanto tal replicação ocorre somente sob certas condições especiais, somente protocolos síncronos bloqueantes são utilizados; o sistema de replicação não é integrado ao sistema de coleta de estados e, por tratar-se de protocolos síncronos, a coleta de lixo é feita de uma forma muito simples.

Nosso objetivo é elaborar algoritmos que estejam integrados ao sistema de coleta de estados, não necessariamente a um protocolo específico, mas que utilize as propriedades existentes nos protocolos de *checkpointing* e que aborde tanto protocolos de *checkpoint* síncronos quanto protocolos quase-síncronos.

Para validar tais algoritmos eles serão implementados utilizando um arcabouço de troca de mensagens baseado no padrão MPI que realiza a coleta e a gravação de estados em disco. Esse arcabouço já está implementado e está sendo adaptado para atender aos requisitos exigidos pelo projeto (como suporte ao envio de *checkpoints* coletados para outros processos, ao retrocesso e à coleta de lixo).

Estágio atual do trabalho e trabalhos futuros: Nesse momento nosso foco é um protocolo síncrono não bloqueante que realize a replicação utilizando mensagens de controle e informações de carona enviadas junto com as mensagens da aplicação. Tal protocolo seria responsável por coordenar a replicação dos estados e pela coleta de lixo das cópias obsoletas.

Em um futuro próximo abordaremos a elaboração de algoritmos de retrocesso para o protocolo de replicação em questão e a implementação de todos os algoritmos propostos para validação desse trabalho. Esses algoritmos serão baseados em protocolos de *checkpointing* [7] e coleta de lixo [8] desenvolvidos por alunos do Laboratório de Sistemas Distribuídos do IC.

Nossa meta é, posteriormente, elaborar um algoritmo semelhante para protocolos quase-síncronos de *checkpointing*, utilizando somente mensagens de controle para realizar o gerenciamento das cópias e a coleta de lixo.

Referências

- [1] R. Y. de Camargo, R. Cerqueira, and F. Kon. Strategies for storage of checkpointing data using non-dedicated repositories on Grid systems. In *MGC '05: Proceedings of the 3rd international workshop on Middleware for grid computing*, pages 1–6, New York, NY, USA, 2005. ACM Press.
- [2] E. N. Elnozahy, D. Johnson, and Y.M.Yang. A Survey of Rollback-Recovery Protocols in Message-Passing Systems. Technical Report CMU-CS-96-181, School of Computer Science, Carnegie Mellon University, 1996.
- [3] D. Manivannan and M. Singhal. Quasi-synchronous checkpointing: Models, characterization, and classification. *IEEE Transactions on Parallel and Distributed Systems*, 7(10):703–713, July 1999.
- [4] J. Plank, J. Xu, and R. Netzer. Compressed differences: An algorithm for fast incremental checkpointing. Technical Report CS-95-302, University of Tennessee, 1995.
- [5] J. S. Plank and K. Li. Faster checkpointing with $n + 1$ parity. Technical report, University of Tennessee, Knoxville, TN, USA, 1993.
- [6] J. S. Plank, K. Li, and M. A. Puening. Diskless Checkpointing. *IEEE Transactions on Parallel and Distributed Systems*, 9(10):972, 1998.
- [7] T. C. Sakata and I. C. Garcia. Non-blocking synchronous checkpointing based on rollback-dependency trackability. In *SRDS '06: Proceedings of the 25th IEEE Symposium on Reliable Distributed Systems (SRDS'06)*, page 411, Washington, DC, USA, 2006. IEEE Computer Society.
- [8] R. M. Schmidt. Coleta de lixo para protocolos de *checkpointing*. Dissertação de mestrado, Instituto de Computação—Universidade Estadual de Campinas, 2003.
- [9] L. Silva and J. Silva. Using two-level stable storage for efficient checkpointing. *Software, IEE Proceedings*, 145:198–201, 1998.
- [10] Y. M. Wang. Consistent Global Checkpoints that Contain a Given Set of Local Checkpoints. *IEEE Trans. on Computers*, 46(4):456–468, Apr. 1997.

21. Criptografia Baseada em Emparelhamentos em Redes de Sensores Sem Fio

Autores: Leonardo B. Oliveira, Ricardo Dahab

Redes de Sensores Sem Fio (RSSFs) [1] são redes ad hoc compostas basicamente por pequenos sensores de recursos limitados (pouca energia, largura de banda, capacidade computacional etc.) e uma ou mais estações rádio base (ERBs), as quais são mais poderosas e conectam os sensores com o ambiente externo. RSSFs são utilizadas com o objetivo de monitorar regiões, oferecendo dados sobre a área monitorada para o resto do sistema. Tais redes podem ser utilizadas para diferentes aplicações, tais como operações de resgate em áreas de conflito e/ou desastre, espionagem industrial e detecção de exploração ilegal de recursos naturais.

Como qualquer outro tipo de rede ad hoc sem fio, RSSFs são vulneráveis a ataques [2]. Porém, além das vulnerabilidades já existentes na comunicação sem fio e em redes ad hoc, RSSFs enfrentam problemas adicionais. Elas comumente são dispostas em ambientes abertos, muitas vezes hostis, o que as tornam fisicamente acessíveis a adversários. Não obstante, nós sensores são mais escassos de recursos que nós de redes ad hoc (o nó sensor Mica2 Motes, por exemplo, possui um processador de 7.8 MHz e 4 KB de memória RAM), e soluções convencionais são não aplicáveis a eles. Por exemplo, o fato de que nós sensores devem ser descartáveis e, por conseguinte, de baixo custo, torna pouco viável equipá-los com dispositivos contra violação (*tamper*). Logo, dotar RSSFs de segurança é uma tarefa especialmente desafiadora.

O baixo poder computacional dos sensores torna inviável a utilização de algoritmos de Criptografia de Chave Pública (*Public Key Cryptography*– PKC) convencionais (RSA/DSA, por exemplo) e, até recentemente, primitivas de segurança como sigilo, autenticação e integridade em RSSFs eram alcançadas através de algoritmos criptográficos simétricos (RC5, SkipJack etc. [3]).

Apesar de mais eficientes, tais algoritmos possuem diversas inconveniências. Primeiro, as partes que desejam se comunicar de forma segura precisam *a priori* decidir por uma chave em comum e compartilhar essa chave através de um canal seguro. Logo, o primeiro desafio é estabelecer esse canal seguro, visto que a necessidade do compartilhamento de chaves advém justamente do desejo de se viabilizar um canal seguro. Este problema é ainda pior em RSSFs pois, ao contrário das redes convencionais em que existem canais alternativos (telefone, correio, múltiplos enlaces etc.) que podem ser usados para a troca de chaves, o único canal existente é o enlace sem fio. A segunda inconveniência é a necessidade de que, para se comunicar de forma segura, é necessário que um indivíduo compartilhe uma chave distinta com cada outro participante da comunicação. Isso resulta em um sério problema de escalabilidade, uma vez que RSSFs em geral são compostas por centenas ou milhares de nós sensores. É verdade que uma opção seria vários nós compartilharem uma mesma chave, mas, nesse caso, a violação de um único nó comprometeria todo o grupo. Fi-

nalmente, o fato de que em algoritmos simétricos pares de indivíduos compartilham uma mesma chave possibilita ataques de refutabilidade (um indivíduo afirmar que não foi ele quem encriptou (*encrypted*) uma mensagem, quando na verdade foi) e personificação (um indivíduo se passar por outro).

A fim de contornar algumas destas inconveniências, diversos trabalhos propuseram o emprego de criptossistemas simétricos por meio de pré-distribuição de chaves ([3], por exemplo). A idéia, em geral, é um compromisso entre conectividade entre os nós e consumo de recursos por partes do mesmo. Contudo, na maioria dos casos, este compromisso não é capaz de fornecer nem perfeita conectividade, nem consumo ideal de recursos.

Atualmente, motivados por sanar tais vulnerabilidades, a comunidade de criptografia em RSSFs vêm investigando o emprego de técnicas mais eficientes de PKC. Uma dessas técnicas é a criptografia baseada em curvas elípticas [4, 5] (*Elliptic Curve Cryptography* – ECC). Utilizando ECC foi mostrado [6] que PKC não é apenas factível em RSSFs, mas uma promessa para a segurança de dispositivos de baixo poder computacional como um todo, pois para um certo nível de segurança, ECC consome muito menos recursos computacionais do que métodos convencionais de PKC.

Todavia, para que essas propostas sejam efetivamente aplicadas em RSSFs, é necessário que as chaves públicas possam ser autenticadas. Caso contrário, ataques do tipo *man-in-the-middle* podem ser disparados. Esse tipo de autenticação é geralmente obtido pelo uso de uma Infra-Estrutura de Chaves Públicas (*Public Key Infra-Structure* – PKI), em que certificados de chaves públicas são emitidos por entidades confiáveis e verificados posteriormente por um usuário interessado. Essas operações, por sua vez, acarretam altos *overheads* de comunicação e computação e, PKIs, portanto, são inadequadas no contexto de RSSFs.

Neste contexto, um evento auspicioso foi o surgimento da Criptografia Baseada em Emparelhamentos (*Pairing-Based Cryptography* – PBC) [7]. PBC é uma nova primitiva criptográfica que vem despertando enorme interesse da comunidade internacional de criptografia, pois possibilita o projeto e utilização de esquemas criptográficos originais e muito elegantes, além de tornar mais eficientes protocolos já conhecidos.

Entre essas aplicações está a Encriptação Baseada em Identidade (*Identity-Based Encryption* – IBE) [8]. IBE foi originalmente proposta por Shamir [9], mas só se tornou viável com o advento de PBC. Diferentemente das demais propostas de PKC, em que PKIs e verificação de certificados são demandados, em IBE uma chave pública é uma identificação conhecida do usuário (seu endereço de correio eletrônico, por exemplo) e por isso é “auto-autenticável”.

Em nosso trabalho, defendemos a idéia de que PBC e RSSFs são complementares. Por um lado, PBC possibilita diversos esquemas criptográficos interessantes no contexto de RSSFs [10]. Por outro, RSSFs atendem bem ao requisitos de PBC [10]. Portanto, propomos implementar primitivas de PBC em nós sensores de baixo poder computacional, bem como projetar protocolos que tirem proveito destas primitivas.

Mais especificamente, nossos macro-objetivos são:

1. mostrar que PBC é, sim, factível em RSSFS, ou seja, implementar eficientemente primitivas de PBC em sensores de recursos limitados; e
2. projetar protocolos criptográficos para RSSFs que tirem proveito dos benefícios de PBC.

Referências

- [1] Deborah Estrin, Ramesh Govindan, John S. Heidemann, and Satish Kumar. Next century challenges: Scalable coordination in sensor networks. In *Mobile Computing and Networking (MobiCom'99)*, p. 263–270, Seattle, 1999.
- [2] C. Karlof and D. Wagner. Secure routing in wireless sensor networks: Attacks and countermeasures. *Elsevier's AdHoc Networks Journal, Special Issue on Sensor Network Applications and Protocols*, 1(2–3):293–315, 2003. Also appeared in 1st IEEE International Workshop on Sensor Network Protocols and Applications.
- [3] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar. SPINS: Security protocols for sensor networks. *Wireless Networks*, 8(5):521–534, 2002.
- [4] V. Miller. Uses of elliptic curves in cryptography, advances in cryptology. In *Crypto'85, Lecture Notes in Computer Science*, volume 218, pages 417–426. Springer-Verlag, 1986.
- [5] N. Koblitz. Elliptic curve cryptosystems. *Mathematics of computation*, 48:203–209, 1987.
- [6] N. Gura, A. Patel, A. Wander, H. Eberle, and S. C. Shantz. Comparing elliptic curve cryptography and rsa on 8-bit cpus. In *Workshop on Cryptographic Hardware and Embedded Systems (CHES)*, p. 119–132, 2004.
- [7] R. Sakai, K. Ohgishi, and M. Kasahara. Cryptosystems based on pairing. In *Symposium on Cryptography and Information Security (SCIS'00)*, p. 26–28, 2000.
- [8] D. Boneh and M. Franklin. Identity-based encryption from the weil pairing. *SIAM J. Comput.*, 32(3):586–615, 2003. also appeared in CRYPTO '01.
- [9] Adi Shamir. Identity-based cryptosystems and signature schemes. In *CRYPTO'84: on Advances in cryptology*, p. 47–53. Springer-Verlag, 1984.
- [10] L. B. Oliveira and R. Dahab. SecLEACH – a random key distribution solution for securing clustered sensor networks. In *5th IEEE International Symposium on Network Computing and Applications*, Cambridge, MA, 2006. fast abstract.

22. Uma Metodologia de Desenvolvimento Rigoroso Centrado na Arquitetura para Sistemas Baseados em Componentes

Autores: Patrick Henrique da Silva Brito e Cecília Mary Fischer Rubira

O avanço da tecnologia computacional disseminou sistemas de software informatizados em várias atividades humanas que são consideradas críticas, como por exemplo, sistemas médicos, de aviação e de controle industrial e financeiro, sendo imprescindível que esses sistemas garantam o seu funcionamento correto. Apesar da exigência por sistemas confiáveis, o mercado de software atual apresenta outros requisitos aparentemente antagônicos, como a necessidade de se desenvolver sistemas de maneira rápida a um custo reduzido. Com o objetivo de satisfazer a esses requisitos, tecnologias como o Desenvolvimento Baseado em Componentes (DBC) [10] e o desenvolvimento centrado na arquitetura [11] vêm sendo cada vez mais utilizadas. Essa adoção é motivada principalmente pela extensibilidade e reutilização de código em um alto nível de granularidade decorrentes dessa utilização [3, 10].

Além disso, com o aumento crescente do tamanho e principalmente da complexidade dos sistemas de software, o projeto arquitetural assumiu um papel decisivo para o sucesso ou falha no atendimento dos requisitos, principalmente dos requisitos não-funcionais, que estão relacionados a aspectos qualitativos do sistema [9]. Por esse motivo, a arquitetura de software deve ter um lugar de destaque no processo de desenvolvimento de sistemas de software complexos [11].

O desenvolvimento centrado na arquitetura apresenta várias vantagens que encorajam a sua adoção. As principais vantagens são: (i) **estruturação abstrata e explícita do sistema**. Essa abstração proporciona um fácil entendimento do sistema e faz da arquitetura um poderoso veículo de comunicação entre todas as suas partes interessadas (do inglês *stakeholders*). (2) **alta granularidade de reutilização**. Essa reutilização estrutural do sistema pode ser determinante para a redução do seu tempo de desenvolvimento, o que atende a um requisito crítico do mercado de software atual [6]. (3) **maior facilidade para adaptar, manter, evoluir e portar o sistema para outras plataformas**. Todas essas facilidades são conseqüências do baixo acoplamento proporcionado pelo uso de conectores arquiteturais, que explicitam a comunicação entre os componentes da arquitetura. Essa representação explícita da comunicação facilita a substituição de componentes do sistema, uma vez que torna possível o pré-processamento das mensagens que fluem entre eles [11].

Em relação à construção de sistemas confiáveis, podem ser utilizadas técnicas de prevenção de falhas (do inglês *fault avoidance*), remoção de falhas (do inglês *fault removal*) ou de tolerância a falhas (do inglês *fault tolerance*). As técnicas de prevenção de falhas visam evitar a inserção de falhas durante o desenvolvimento do sistema [9]; por isso, essas técnicas normalmente são alcançadas através da utilização de processos rigorosos de desenvolvimento. Enquanto isso, as técnicas de remoção de falhas têm o intuito de reduzir o número de falhas residuais do sistema, antes de

implantá-lo no ambiente do usuário final. Finalmente, de forma complementar às outras técnicas, as técnicas de tolerância a falhas visam conviver amistosamente com a presença de falhas, mantendo o sistema em funcionamento, mesmo nessas situações indesejadas [1]. A forma mais utilizada atualmente para implementar tolerância a falhas é através de mecanismos de tratamento de exceções [2].

Tratamento de exceções [5] é uma técnica bastante conhecida para a estruturação do comportamento excepcional em sistemas de software. Além disso, é implementado por diversas linguagens de programação populares, como C++, Java e C#.

Apesar dessa popularidade, o projeto e a implementação do comportamento excepcional de um sistema são tarefas muito complexas que não recebem a atenção devida de processos de desenvolvimento existentes [4,8]. A situação é ainda mais crítica se levarmos em consideração os métodos para DBC e mais ainda, se considerarmos os processos centrados na arquitetura. Conseqüentemente, os desenvolvedores sentem dificuldade tanto para usar os mecanismos de tratamento de exceções, quanto para projetar o comportamento excepcional, o que compromete a confiabilidade dos sistemas construídos [7].

Este trabalho propõe um processo centrado na arquitetura para o desenvolvimento de sistemas robustos. O processo proposto deve abranger todo o ciclo de vida do software, isto é, desde a especificação de requisitos, passando pelas fases de desenvolvimento e atuando inclusive na fase de manutenção do sistema. Além disso, o processo deve ser ajustável de acordo com a criticidade da aplicação. Nesse sentido, deve ser possível adotar as técnicas de prevenção, remoção, e tolerância a falhas de forma isolada ou combinada. As contribuições inéditas do trabalho encontram-se principalmente no que tange à sistemática no desenvolvimento de arquiteturas de software robustas com verificação formal de suas respectivas propriedades, além do uso combinado das técnicas de prevenção, remoção e tolerância a falhas. Além disso, outro diferencial do trabalho proposto é a preocupação com o tratamento de exceções de forma coordenada entre vários componentes do sistema, que será um requisito determinante durante a especificação da arquitetura de software. No que se refere a tolerância a falhas, o projeto se baseará num método desenvolvido no Instituto de Computação da Unicamp [2].

O processo proposto também deve apoiar o desenvolvimento de sistemas críticos em linhas de produtos de software, onde a reutilização ocorre em um nível de granularidade ainda maior. Nesse sentido, será definido um meta-modelo para estruturar a especificação de variabilidades arquiteturais, relacionadas ao comportamento excepcional de sistemas baseados em componentes.

Referências

- [1] Thomas Anderson and Peter A. Lee. *Fault Tolerance: Principles and Practice*. Prentice-Hall, 2nd edition, 1990.
- [2] Patrick H. S. Brito, Fernando Castor Filho, and Cecília M. F. Rubira. Um método para modelagem de exceções em desenvolvimento baseado em componentes. *IV Workshop de Desenvolvimento Baseado em Componentes*, August 2004.
- [3] John Chessman and John Daniels. *UML Components: A Simple Process for Specifying Component-Based Software*. Paperback, 1992.
- [4] Rogério de Lemos and A. Romanovsky. Exception handling in a cooperative object-oriented approach. In *Proc. of the 2nd IEEE ISORC'99*, May 1999.
- [5] John B. Goodenough. Exceptional handling: Issues and a proposed notation. *CACM*, 18(12), 1975.
- [6] Brian Randell. Dependable pervasive systems. *23rd IEEE International Symposium on Reliable Distributed Systems (SRDS'04)*, October 2004.
- [7] Darrel Reimer and Harini Srinivasan. Analysing exception usage in large java applications. In *Proc. of ECOOP Workshop on Exception Handling in Object-Oriented Systems (EHOOS'2003)*, 2003.
- [8] C. Rubira, R. de Lemos, and G. Ferreira e F. Castor Filho. Exception handling in the development of dependable component-based systems (to appear). In *Software Practice and Experience*, 2004.
- [9] Ian Sommerville. *Software Engineering*. Addison-Wesley, 5th edition, 1995.
- [10] Clemens Szyperski. *Component Software: Beyond Object-Oriented Programming*. Addison-Wesley, 1998.
- [11] Zhang You-Sheng and He Yu-Yun. Architecture-based software process model. *Software Engineering Notes*, 28(2), March 2003.

23. Algoritmos para Problemas de Particionamento em Grafos

Autores: Evandro Cesar Bracht, Flávio K. Miyazawa e Orlando Lee

O foco do nosso trabalho consiste em estudar Problemas de Particionamento em Grafos. Os problemas abordados são NP-completos e portanto, não existem algoritmos exatos com complexidade de tempo polinomial para esses, supondo que $P \neq NP$. Sendo assim, uma série de abordagens são consideradas na pesquisa, desde algoritmos aproximativos e heurísticas a algoritmos exatos. Os resultados podem ser algoritmos rápidos, com potencial aplicação a problemas reais, ou algoritmos polinomiais sem aplicação direta, mas que trazem mudanças nos atuais limites de aproximação.

Problemas de Particionamento em Grafos são de grande interesse para a comunidade científica. Eles têm uma enumerável gama de funcionalidades em Computação Aplicada. Todo problema que possa ser descrito por meio de um problema em grafos é um potencial candidato a utilizar esses algoritmos.

Um dos problemas de particionamento estudado é o problema *Correlation Clustering* (CC), que assim como outros problemas de particionamento, tem como objetivo criar uma partição onde objetos semelhantes estejam juntos na mesma parte enquanto que objetos não semelhantes pertençam a partes diferentes.

No problema CC é dado um conjunto V de n documentos, que desejamos separar por tópicos, e para cada dois documentos u e v em V temos a informação se estes são similares ou diferentes, e o nosso objetivo é encontrar um particionamento que concorde ao máximo com a informação de similaridade. Podemos interpretar este problema como sendo um problema em grafos, onde é dado um grafo completo $G = (V, E)$, onde as arestas em $E = E^+ \cup E^-$ são rotuladas com “+” (similares) ou “-” (diferentes), e o objetivo é encontrar um particionamento que concorde ao máximo com o rótulo das arestas.

O objetivo de “concordar ao máximo com o rótulo das arestas” pode ser visto de duas formas, minimizar os erros ou maximizar os acertos. Dado um particionamento $\mathcal{C} = \{C_1, \dots, C_m\}$ do grafo G , com $1 \leq m \leq n$, $C_i \cap C_j = \emptyset$ para todo $i \neq j$ e $\bigcup_{i=1}^m C_i = V$, o conjunto de **erros** é formado pelas arestas $uv \in E^+$, tal que $u \in C_i$, $v \notin C_i$, e pelas arestas $uv \in E^-$, tal que $u, v \in C_i$. Ou seja, os erros são as arestas rotuladas com “-” internas as partes e as arestas rotuladas com “+” entre partes. Já o conjunto dos **acertos** para um particionamento \mathcal{C} é formado pelas arestas $uv \in E^+$, tal que $u, v \in C_i$, e pelas arestas $uv \in E^-$, tal que $u \in C_i, v \notin C_i$. Ou seja, os acertos são as arestas rotuladas com “+” internas as partes e as arestas rotuladas com “-” entre partes.

Baseado nos acertos e nos erros resultantes de um particionamento, consideramos duas funções objetivos distintas para o problema CC.

Problema MINDISAGREE: *Dado um grafo $G = (V, E)$ com rótulos “+” e “-” nas arestas, encontre um particionamento \mathcal{C} , de modo que minimize os erros.*

Problema MAXAGREE: *Dado um grafo $G = (V, E)$ com rótulos “+” e “-” nas arestas, encontre um particionamento \mathcal{C} que maximize os acertos.*

Note que estes dois problemas são equivalentes na solução ótima, mas são diferentes no ponto de vista de aproximação. Para o problema MAXAGREE em grafos completos existe um PTAS apresentado por Bansal *et al.* [2], já para o problema MINDISAGREE o melhor resultado é uma 2,5 aproximação, dada por Ailon *et al.* [1].

Na literatura encontramos algumas variações para os problemas MAXAGREE e MINDISAGREE. Estas variações se devem ao fato do grafo ser completo ou não, e na função de pesos das arestas. Podemos dizer que no caso em que o grafo não é completo os problemas MAXAGREE e MINDISAGREE se tornam mais difíceis. Para o problema MAXAGREE o melhor resultado existente na literatura é um algoritmo 0,7664-aproximação que utiliza programação semidefinida apresentado por Charikar *et al.* [3]. Já para o problema MINDISAGREE o melhor resultado é uma $\log n$ -aproximado apresentado simultaneamente por Demaine e Immorlica [4], Emanuel e Fiat [5] e Charikar *et al.* [3]. Estes autores também apresentam uma prova de equivalência entre o problema MINDISAGREE em grafos gerais e o problema Multicut.

Baseado no problema de Classificação Métrica [8, 9] propusemos uma nova variação para o problema CC, denominada de Problema *Labeled Correlation Clustering* (LCC). No problema LCC além do grafo completo $G(V, E)$ com rótulos nas arestas temos um conjunto de classes L e a informação se o objeto u “gostaria” de ser associado a classe i , esta informação é dada através de rótulos “+” (gostaria) ou “-” (não gostaria) na relação $V \times L$. O objetivo do problema LCC é encontrar um particionamento rotulado $\mathcal{C}^L = \{C_i\}$, com $i \in L$, $C_i \cap C_j = \emptyset$ e $\bigcup_{i \in L} C_i = V$.

Assim como no problema CC uma aresta uv , com $u, v \in V$, é um acerto se esta é rotulada com “+” e $u, v \in C_i$ ou é rotulada com “-” e $u \in C_i$, $v \notin C_i$, além destes temos os acertos que ocorrem quando associamos um objeto u a uma classe i e a relação ui é rotulada com “+”, para todo $u \in V$ e $i \in L$. Já a aresta uv , com $u, v \in V$, é um erro se esta é rotulada com “-” e $u, v \in C_i$ ou é rotulada com “+” e $u \in C_i$, $v \notin C_i$, além destes temos os erros que ocorre quando associamos um objeto u a uma classe i e a relação ui é rotulada com “-”. Note que não consideramos os acertos nas arestas rotuladas com “-” e os erros das arestas rotuladas com “+” na relação $V \times L$, pois se o fizermos teremos o problema CC em grafos gerais. Baseados nos acertos e erros temos duas funções objetivo para o problema LCC, maximizar o numero de acertos (MAXAGREEL) ou minimizar o número de erros (MINDISAGREEL).

Ao fixarmos o número de classe para o problema MINDISAGREEL obtemos um PTAS usando a idéia apresentada por Giotis e Guruswami [7] para o problema MINDISAGREE[k]. Além disto temos uma formulação em Programação Linear (PL) para o problema MINDISAGREEL e estamos investigando um algoritmo para transformar a solução fracionária, devolvida pela PL, em uma solução inteira. Estamos interessados também em instâncias reais para a aplicações do problema LCC, assim como resultados de inaproximabilidade.

Referências

- [1] N. Ailon A. Newman, M. Charikar. Aggregating inconsistent information: Ranking and clustering. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing (STOC)*, volume 14B, pages 684–693, 2005.
- [2] N. Bansal, A. Blum, and S. Chawla. Correlation clustering. In *Proc. of the 43rd IEEE Symposium on Foundations of Computer Science*, page 238, 2002.
- [3] M. Charikar, V. Guruswami, and A. Wirth. Clustering with qualitative information. In *Proceedings of of the 44th Annual IEEE Symposium on Foundations of Computer Science*, page 524. IEEE Computer Society, 2003.
- [4] E. Demaine and N. Immorlica. Correlation clustering with partial information. In *Proceedings of the 6th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems and 7th International Workshop on Randomization and Approximation Techniques in Computer Science*, pages 1–13, 2003.
- [5] D. Emanuel and A. Fiat. Correlation clustering - minimizing disagreements on arbitrary weighted graphs. In *Proceedings of the 11th Annual European Symposium*, volume 2832 of *Lecture Notes in Computer Science*, pages 208–220, 2003.
- [6] A. Gionis, H. Mannila, and P. Tsaparas. Clustering aggregation. In *21st International Conference on Data Engineering (ICDE'05)*, pages 341–352, 2005.
- [7] I. Giotis and V. Guruswami. Correlation clustering with a fixed number of clusters. In *Proc. of 17th annual ACM-SIAM symposium on Discrete algorithm*, pages 1167–1176, 2006.
- [8] A. Gupta and E. Tardos. Constant factor approximation algorithms for a class of classification problems. In *Proceedings of the 32nd Annual ACM Symposium on the Theory of Computing*, pages 125–131, 1998.
- [9] J. Kleinberg and E. Tardos. Approximation algorithms for classification problems with pairwise relationships: Metric labeling and markov random fields. In *Proceedings of the 40th Annual IEEE Symposium on Foundations of Computer Science*, pages 14–23, 1999.
- [10] C. Swamy. Correlation clustering: maximizing agreements via semidefinite programming. In *Proceedings of the 15th annual ACM-SIAM Symposium on Discrete algorithms*, pages 526–527. Society for Industrial and Applied Mathematics, 2004.

24. The 2D-VLIW Architecture: Execution Model and Code Generation

Authors: Ricardo Santos, Rodolfo Azevedo, and Guido Araujo

For many years the constant increase in processor performance came from the advancement in VLSI process technology. However, recent announcements on the technology limits due to the thermal barrier have motivated the research into innovative ways to sustain the increase in performance.

One approach used by modern computer architectures to achieve performance, independent of the implementation technology, is the addition of hardware resources to exploit the parallelism available inside programs. Two architectural design styles commonly used to exploit parallelism are VLIW (Very Long Instruction Word) and superscalar. Despite all their advantages and disadvantages, the major problem with VLIW and superscalar architectures is related to its inability to capture and scale parallelism. Moreover, previous work [3, 8] as well as state-of-the-art architectures [1, 4] have pointed out towards architectures adopting new strategies for code optimization and execution models. Advanced compiler optimizations could look at large code regions to find out greater parallelism levels. The execution models could provide new resource arrangements to execute large code regions, thus increasing performance.

In this work, we propose a new architecture and execution model where the main characteristic is the mapping of large pieces of computation into a matrix of functional units. This mapping brings attractive performance to applications with high ILP as embedded and some general-purpose applications. Our architecture is named 2D-VLIW [5, 6] because large instructions, comprised of single operations, are fetched from the memory and executed into a (two-dimensional) matrix of functional units connected through a narrow interconnection network. The code generation to this architecture is particularly interesting due to the features and constraints of the matrix. We show how single operations compose large 2D-VLIW instructions and how these instructions are scheduled onto the architecture by using a set of algorithms and heuristics based on the subgraph isomorphism concept.

In order to evaluate the real performance gains from the 2D-VLIW architecture, we have compared it to an EPIC-based processor [7] using Mediabench and SPEC suites compiled on the Trimaran compiler [2]. We assume that an EPIC processor has more flexibility in terms of operations (dependent and parallel) within an instruction and hardware complexity than a pure VLIW processor. The experiments revealed speedups ranging from 5%-63% over the EPIC processor and, for some programs, the average 2D-VLIW OPC (operations per cycle) was up to 10 times greater than for the EPIC processor.

The 2D-VLIW architecture exploits instruction level parallelism through a pipelined datapath and a two dimensional VLIW execution model. This architecture fetches large instructions from the memory as well as VLIW and EPIC architectu-

res do. The 2D-VLIW instructions are comprised of single operations executed by a set of functional units (FUs) organized as a matrix. The maximum number of operations in one instruction is equivalent to the amount of functional units in the matrix. Figure 3 shows a simplified overview of a 4×4 2D-VLIW datapath as well as the content of a matrix functional unit.

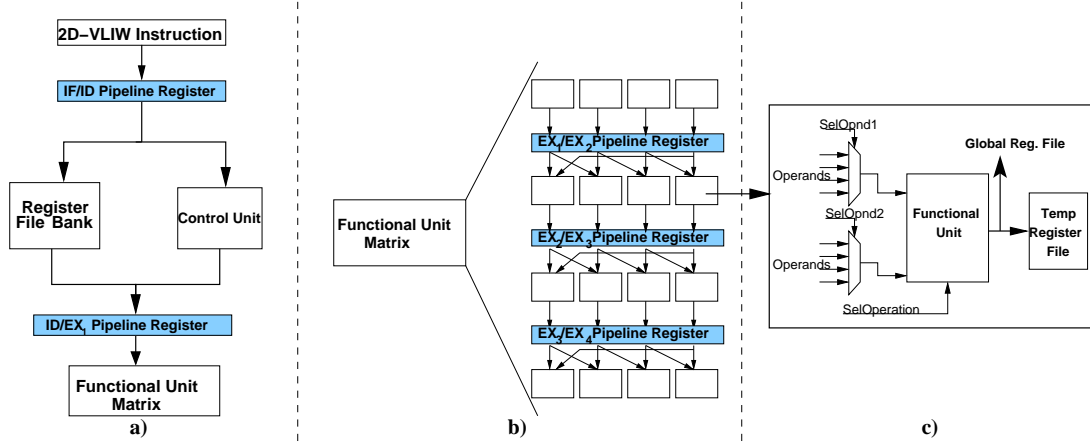


Figure 3: A simplified overview of the 2D-VLIW datapath.

In Figure 3(b) four operations are executed by 4 FUs at each execution stage EX_1, EX_2, EX_3, EX_4 of the pipeline. Notice that, by using the interconnection network, each FU in row n , $1 \leq n < 4$, can provide operands to two FUs in the following row $(n+1)$. Figure 3(c), on the other hand, shows all logic blocks inside an FU. Results from an FU may be written either into the *Temp Register File* (TRF) or into the *Global Register File* (GRF). TRF is a small register bank with 2 local registers dedicated to each FU. The result from an operation is always written into an internal register called *FU Register*.

The TRFs in each FU are a viable alternative to minimize the pressure over the global register file. This is achieved by storing temporary values (of the Directed Acyclic Graphs - DAGs) into TRFs. The DAG leaves' read values from the global register file, while the operations inside the DAGs read and write values from/to the temporary register. The DAG root nodes write values into the global register file.

The 2D-VLIW instruction scheduling is based on the subgraph isomorphism concept. In this scheduling program DAGs' are matched to the architecture's graph. The architecture graph is comprised of the matrix of functional units. The subgraph isomorphism problem is *NP*-complete [9] and due to this, we have adopted the VF subgraph isomorphism library [10] and a set of heuristics in order to perform the scheduling of instructions on the 2D-VLIW architecture. Our preliminary results show that the subgraph isomorphism strategy overcomes list scheduling and modulo scheduling algorithms in terms of application final latency and instruction fulfillment.

Our performance results show that the combination of architectural arrangement, execution model, and compiler assistance can achieve an appealing performance to embedded applications. The compiler plays an important role in the 2D-VLIW architecture. It is responsible for recognizing large pieces of computation, mapping their operations onto the matrix of functional units and using local registers to keep temporary values. Currently, we are working on algorithms to coding large instructions, extracting patterns from these instructions, and register allocation algorithms taking the 2D-VLIW arrangement into account.

Referências

- [1] D. Burger, S. W. Keckler, K. S. McKinley, M. Dahlin, L. K. John, C. Lin, C. R. Moore, J. Burril, R. G. McDonald, and W. Yoder. Scaling to the End of Silicon with EDGE Architectures. *IEEE Computer*, 37(7):44–55, 2004.
- [2] L. N. Chakrapani, J. Gyllenhaal, W. Mei, W. Hwu, S. A. Mahlke, K. V. Palem, and R. M. Rabbah. Trimaran - An Infrastructure for Research in Instruction-Level Parallelism. *Lecture Notes in Computer Science*, 3602:32–41, 2004.
- [3] M. S. Lam and R. P. Wilson. Limits of Control Flow on Parallelism. In *Proceedings of the 19th International Symposium on Computer Architecture (ISCA)*, pages 46–57. ACM Press, 1992.
- [4] B. Mei, A. Lambrechts, J.-Y. Mignolet, D. Verkest, and R. Lauwereins. Architecture Exploration for a Reconfigurable Architecture Template. *IEEE Design & Test of Computers*, 22(2):90–101, 2005.
Society.
- [5] R. Santos, R. Azevedo, and G. Araujo. The 2D-VLIW Architecture. Technical Report IC-06-006, Institute of Computing, 2006.
- [6] R. Santos, R. Azevedo, and G. Araujo. 2D-VLIW: An Architecture Based on the Geometry of the Computation. In *Proceedings of the 17th IEEE International Application-specific Architectures, Processors and Systems Conference (ASAP)*, SteamBoat Springs-CO-USA, 2006. IEEE Computer Society.
- [7] M. S. Schlansker and B. R. Rau. EPIC: Explicitly Parallel Instruction Computing. *IEEE Computer*, 33(2):37–45, 2000.
- [8] D. W. Wall. Limits of Instruction-Level Parallelism. In *Proceedings of the 4th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, pages 176–188. ACM Press, 1991.
- [9] M. R. Garey, D. S. Johnson. Computers and Intractability: A Guide to the Theory of NP-Completeness. W. H. Freeman and Company, San Francisco-CA-USA, 1979.
- [10] L. P. Cordella, P. Foggia, C. Sansone, M. Vento. Evaluating Performance of the VF Graph Matching Algorithm. In *Proceedings of the 10th International Conference on Image Analysis and Processing*, pages 1172–1177. IEEE Computer Society Press, 1999.

25. Uma Ponte entre as Abordagens Síncrona e Quase-síncrona para Checkpointing

Autores: Tiemi Christine Sakata e Islene Calciolari Garcia

Checkpointing é um mecanismo utilizado para garantir tolerância a falhas em sistemas distribuídos. Um protocolo de *checkpointing* é responsável pelo armazenamento de estados em memória estável, chamados de *checkpoints*, que podem ser utilizados para o restabelecimento de um processo falho. Após a ocorrência de uma falha é necessário que o sistema retroceda para o último estado global consistente, ou seja, um estado global que poderia ter sido obtido por um observador onisciente externo [3]. Os protocolos de *checkpointing* podem ser classificados como assíncronos, quase-síncronos ou síncronos.

Na abordagem assíncrona, os processos têm autonomia na escolha de seus *checkpoints*, porém não há garantias de que seja possível formar um *checkpoint* global consistente a partir dos *checkpoints* salvos. Em caso de falha, a aplicação pode ser obrigada a retornar ao estado inicial (efeito dominó) [6].

Os protocolos quase-síncronos (ou induzidos por comunicação) propostos na literatura reduzem ou eliminam o número de *checkpoints* inúteis, induzindo o armazenamento de *checkpoints* adicionais aos *checkpoints* do protocolo assíncrono [4]. Nesta abordagem, os processos salvam seus *checkpoints* livremente, chamados *checkpoints* básicos e podem ser induzidos a armazenarem *checkpoints* adicionais, chamados *checkpoints* forçados, segundo predicados avaliados sobre informações de controle propagadas via *piggybacking* pelas próprias mensagens da aplicação.

Em contraste com as duas classes de protocolos já mencionadas, os protocolos síncronos constroem *checkpoints* globais consistentes utilizando fases de troca de mensagens de controle que sincronizam processos sinalizando o momento de início e término da execução do protocolo. Ao final de uma execução do protocolo síncrono, o *checkpoint* global consistente representado pelos últimos *checkpoints* locais é o *checkpoint* global consistente mais recente existente no sistema.

O alto número de mensagens de controle somado ao alto custo do armazenamento de *checkpoints* gerado por estes protocolos motivou o desenvolvimento de protocolos que reduzem o número de mensagens de controle e o número de *checkpoints* salvos durante uma execução de *checkpointing*. Chamamos de protocolos minimais os protocolos que determinam o conjunto minimal de processos que devem gravar um *checkpoint* para garantir a construção de um *checkpoint* global consistente [8, 9]. Cao e Singhal demonstraram que todo protocolo minimal precisa ser bloqueante (processos envolvidos suspendam suas atividades da computação sob o ponto de vista da aplicação) [1]. Porém, a suspensão da aplicação reduz o desempenho do sistema.

Os protocolos síncronos não-bloqueantes permitem que todos os processos gravem *checkpoints* a cada execução de *checkpointing* sem suspender as atividades da computação. Cao e Singhal propuseram um novo tipo de *checkpoint*, chamado de

checkpoint mutável, que é facilmente manipulado pois pode ser salvo em memória não-estável [2]. Os *checkpoints* mutáveis são armazenados para evitar a formação de *checkpoints* globais inconsistentes e são transferidos para memória estável apenas quando são necessários para uma execução de *checkpointing*, garantindo assim a minimalidade no número de *checkpoints* em memória estável (*checkpoints* estáveis). Estes protocolos são baseados em protocolos minimais e são muito complexos, pouco eficientes e permitem apenas uma execução de *checkpointing* a cada instante de tempo.

Até este momento, os protocolos quase-síncronos e síncronos propostos na literatura foram desenvolvidos e analisados utilizando modelos distintos. Este trabalho propõe o uso dos conceitos e da teoria existentes para os protocolos quase-síncronos para analisar e desenvolver protocolos síncronos. Desta forma, obtivemos as seguintes contribuições:

- Verificação da relação de continência do conceito de z-dependência proposto para os protocolos síncronos [1] na relação de z-precedência utilizada pelos protocolos quase-síncronos. Assim, adotamos o conceito de z-precedência na maioria das provas descritas nesta tese.
- Prova de que protocolos minimais que anotam a recepção de mensagens, mas não os intervalos de origem dessas mensagens não têm informação suficiente para garantir minimalidade no número de *checkpoints*. Mostramos que os protocolos propostos em [2,5] não são minimais e propomos dois novos protocolos minimais, chamados de VD-minimal [8] e Broad-minimal [9] para corrigir os problemas encontrados.
- Prova da impossibilidade de um protocolo síncrono não-bloqueante garantir um número mínimo de *checkpoints* estáveis durante toda a execução da aplicação mesmo na presença de um único iniciador a cada instante.
- Prova da impossibilidade de um protocolo síncrono não-bloqueante salvar um número minimal de *checkpoints* na presença de iniciadores concorrentes.
- Proposta de dois novos protocolos síncronos não-bloqueantes baseados em protocolos quase-síncronos: RDT-NBS [7] e BCS-NBS. Estes protocolos foram baseados em classes distintas de protocolos quase-síncronos e obtivemos como resultado protocolos mais simples que os existentes e que permitem iniciadores concorrentes.
- Resultados de simulação obtidos com o uso do ChkSim [10] que contrastam o comportamento dos protocolos quase-síncronos FDAS e BCS e suas respectivas versões síncronas não-bloqueantes RDT-NBS e BCS-NBS.

Nesta mesma linha de pesquisa, alguns pontos ainda podem ser investigados. Outros protocolos síncronos não-bloqueantes baseados em diferentes protocolos quase-síncronos poderiam ser desenvolvidos para analisar e avaliar o desempenho e custo

desses protocolos. Além disso, o simulador poderia ser modificado para permitir também os protocolos síncronos bloqueantes. Acreditamos que se desacoplarmos a detecção da terminação em protocolos síncronos não-bloqueantes e implementarmos um protocolo de coleta de lixo, poderemos obter protocolos mais simples e com menor número de mensagens de controle.

Referências

- [1] G. Cao e M. Singhal. On the Impossibility of Min-process Non-blocking Checkpointing and an Efficient Checkpointing Algorithm for Mobile Computing Systems. Em *Proc. 27th International Conference on Parallel Processing*, pp. 37–44, New York, 1998. IEEE Press.
- [2] G. Cao e M. Singhal. Checkpointing with Mutable Checkpoints. *Theoretical Computer Science*, 290(2):1127–1148, jan 2003.
- [3] M. Chandy e L. Lamport. Distributed Snapshots: Determining Global States of Distributed Systems. *ACM Transaction on Computing Systems*, 3(1):63–75, fev. 1985.
- [4] D. Manivannan e M. Singhal. Quasi-Synchronous Checkpointing: Models, Characterization, and Classification. Em *IEEE Transaction on Parallel and Distributed Systems*, volume 10, pp. 703–713, julho 1999.
- [5] R. Prakash e M. Singhal. Minimal Global Snapshot and Failure Recovery using Infection. Relatório Técnico OSU-CISRC-12/93-TR42, Department of Computer Science, The Ohio State University, 1993.
- [6] B. Randell. System Structure for Software Fault Tolerance. *IEEE Transaction on Software Engineering*, 1(2):220–232, junho 1975.
- [7] T. C. Sakata e I. C. Garcia. Non-Blocking Synchronous Checkpointing Based on Rollback-Dependency Trackability. Em *25th IEEE Symposium on Reliable Distributed Computing Systems*, Leeds, UK, out. 2006.
- [8] T. C. Sakata, I. C. Garcia e L. E. Buzato. Checkpointing Síncrono Bloqueante Minimal com Iniciadores Concorrentes. Em *Simpósio Brasileiro de Redes de Computadores*, pp. 681–696, Natal, Rio Grande do Norte, maio 2003.
- [9] T. C. Sakata, I. C. Garcia e L. E. Buzato. Uso de Broadcast na Sincronização de Checkpoints em Protocolos Minimais. Em *V Workshop de Testes e Tolerância a Falhas*, pp. 153–164, Gramado, Rio Grande do Sul, maio 2004.
- [10] G. M. D. Vieira e L. E. Buzato. Distributed Checkpointing: Analysis and Benchmarks. Em *Simpósio Brasileiro de Redes de Computadores*, Curitiba, Paraná, maio 2006.

26. Coloração de Arestas: o Problema da Classificação

Autores: Sheila de Almeida e Célia P. de Mello

Neste trabalho, consideramos $G = (V(G), E(G))$ um grafo simples com um conjunto de vértices $V(G)$ e um conjunto de arestas $E(G)$ e $n = |V(G)|$ e $m = |E(G)|$. O grau de um vértice v no grafo G será denotado por $d_G(v)$. Um $\Delta(G)$ -vértice é um vértice de $V(G)$ com grau $\Delta(G)$, onde $\Delta(G)$ é o grau máximo de G . Um vértice universal é um vértice com grau $n-1$. Uma clique é um subconjunto de vértices dois a dois adjacentes em G . Um conjunto independente é um subconjunto de vértices dois a dois não adjacentes em G .

Uma k -coloração de arestas de um grafo G é uma atribuição de k cores para as arestas de $E(G)$ tal que as arestas incidentes em um mesmo vértice têm cores distintas. O índice cromático de G é o menor k para o qual G possui uma k -coloração e é denotado por $\chi'(G)$. Desta definição, podemos concluir que $\chi'(G) \geq \Delta(G)$.

Em 1964, Vizing [9] mostrou que para qualquer grafo G , $\chi'(G) \leq \Delta(G) + 1$, dando origem ao conhecido *Problema da Classificação* que consiste em decidir, dado um grafo G , se $\chi'(G) = \Delta(G)$ ou $\chi'(G) = \Delta(G) + 1$. No primeiro caso, dizemos que G pertence à *Classe 1*, caso contrário, G pertence à *Classe 2*. Apesar de a poderosa restrição imposta por Vizing, é muito difícil computar o índice cromático de um grafo G qualquer. Decidir se G pertence à Classe 1 é NP-completo [6].

O conceito de grafo overfull pode ser usado para classificar alguns grafos que pertencem à Classe 2. Um grafo G é *overfull* (\mathcal{O}) quando n é ímpar e $\Delta(G) \lfloor \frac{n}{2} \rfloor < m$ [5, 8]. Um grafo G é *subgrafo-overfull* (\mathcal{SO}) quando possui um subgrafo H com $\Delta(H) = \Delta(G)$ e H é overfull [5]. Se o subgrafo-overfull H for induzido por um $\Delta(G)$ -vértice e todos os seus vizinhos, dizemos que G é *vizinhança-overfull* (\mathcal{NO}) [2, 3]. Essas classes estão relacionadas entre si da seguinte forma: $\mathcal{O} \subset \mathcal{SO} \subset$ Classe 2 e $\mathcal{NO} \subset \mathcal{SO} \subset$ Classe 2 e \mathcal{O} e \mathcal{NO} não são comparáveis.

Uma forma de classificar um grafo G como Classe 1 é exibir uma $\Delta(G)$ -coloração das arestas de G . Os grafos completos com grau máximo par, os bipartidos, os duplamente cordais com grau máximo ímpar e os split com grau máximo ímpar são exemplos de grafos pertencentes à Classe 1 [1, 2, 4].

Nosso interesse é a coloração de arestas de grafos split. Um grafo G é *split* se o conjunto dos vértices de G pode ser particionado em um conjunto independente e uma clique. O problema da classificação permanece aberto para esta classe, embora esteja resolvido para algumas subclasses [1, 7].

Considere um grafo G com vértice universal. O grafo G é subgrafo de um grafo completo K_n . Logo, se n é par, G pertence à Classe 1. Essa observação e o Lema 1 classificam os grafos que possuem vértice universal.

Lema 1 [8] *Seja G um grafo com vértice universal e n ímpar. Então G pertence à Classe 1 se, e somente se, G não é overfull.*

Seja G um grafo split sem vértice universal. Considere a partição $\{Q, S\}$ de $V(G)$, onde Q é uma clique maximal de G e S é um conjunto independente. O subgrafo induzido pelo conjunto de vértices Q será denotado por K_q , onde $q = |Q|$. Associamos a G um grafo bipartido B , com partição $\{Q, S\}$, obtido de G pela remoção de todas as arestas de K_q . Denotamos por $d(Q) = \max\{d_B(v), \forall v \in Q\}$ e por $d(S) = \max\{d_B(v), \forall v \in S\}$. Portanto, $\Delta(B) = \max\{d(Q), d(S)\}$ e $\Delta(G) = d(Q) + \Delta(K_q)$.

Há dois casos: $\Delta(B) = d(Q)$ ou $\Delta(B) > d(Q)$. Primeiro, considere $\Delta(B) = d(Q)$ e $d(Q) \geq d(S)$. Chen, Fu, e Ko [1] mostraram que os grafos split que possuem todos os $\Delta(B)$ -vértices em Q pertencem à Classe 1. Nesse caso, $d(Q) > d(S)$. Nós provamos que G também pertence à Classe 1 quando existem $\Delta(B)$ -vértices em ambos os conjuntos Q e S .

Lema 2 *Seja G um grafo split. Se $d(Q) = d(S)$, então G pertence à Classe 1.*

Os resultados apresentados acima solucionam completamente o caso $\Delta(B) = d(Q)$. Considere $\Delta(B) > d(Q)$. Note que, neste caso, os $\Delta(B)$ -vértices estão todos em S .

Chen, Fu e Ko [1] mostraram que grafos com grau máximo ímpar pertencem à Classe 1. Lembre-se que, pelo Lema 1, se G tem um vértice universal, então G pertence à Classe 2 se, e somente se, G é \mathcal{SO} . Em [3], Figueiredo et al. mostraram que $\mathcal{NO} = \mathcal{SO}$ para grafos split. Portanto, desejamos resolver o problema da classificação para grafos split que não são vizinhança-overfull, tem grau máximo par, $d(S) > d(Q)$ e não possuem vértice universal.

Nosso resultado é apresentado no Teorema 3 e está fortemente baseado na coloração construída por Planthold [8] para a demonstração do Lema 1. Seja H um grafo com vértice universal, que não é overfull e $|V(H)|$ ímpar. Para mostrar que H pertence à Classe 1, Planthold particiona $V(H)$ em dois conjuntos L e R tal que existe uma $\Delta(H)/2$ -coloração para $H[L]$ e $H[R]$ usando as mesmas $\Delta(H)/2$ cores. Depois, ele mostra como colorir com $\Delta(H)/2$ cores novas o grafo bipartido com partição $[L, R]$.

Consideramos um dos subcasos da demonstração de Planthold onde a partição $[L, R]$ de $V(H)$ satisfaz as seguintes condições: toda aresta de \overline{H} tem um extremo em L ; todo vértice com grau pelo menos dois em \overline{H} pertence à L ; $|L| = (\Delta(H)/2) + 1$ e $R = V(H) \setminus L$, e mostramos que é possível estender a coloração de H a uma subclasse dos grafos split.

Teorema 3 *Seja G um grafo split que não é vizinhança-overfull e tem $\Delta(G)$ par. Se existe um subgrafo $H = G[N[v]]$, onde v é um $\Delta(G)$ -vértice, e $V(H)$ tem uma partição $[L, R]$ tal que:*

- *toda vértice de $V(H)$ com grau pelo menos dois em \overline{H} pertence à L ;*
- *toda aresta de \overline{H} tem um extremo que é vértice de L ;*

- para todo vértice $w \in L$, $N[w] \cap S \subset L$, onde S é o conjunto independente de G ;
- $|L| = (\Delta(H)/2) + 1$;
- $R = V(H) \setminus L$;

então G pertence à Classe 1.

Referências

- [1] B-L. Chen, H-L. Fu e M. T. Ko. Total chromatic number and chromatic index of split graphs. *J. Combin. Maths. Combin. Comp.*, 17:137–146, 1995.
- [2] C. M. H. de Figueiredo, J. Meidanis e C. P. de Mello. A greedy method for edge-colouring odd maximum degree doubly chordal graphs. *Congr. Numer.*, 111:170–176, 1995.
- [3] C. M. H. de Figueiredo, J. Meidanis e C. P. de Mello. Local conditions for edge-colouring. *Journal of Combinatorial Mathematics and Combinatorial Computing*, 32:79–91, 2000.
- [4] S. Fiorini e R. J. Wilson. *Edge-colourings of graphs*. Research Notes in Mathematics, Pitman, 1977.
- [5] A. J. W. Hilton. Two conjectures on edge-colouring. *Discrete Mathematics*, 74:61–64, 1989.
- [6] I. Holyer. The NP-completeness of edge-coloring. *SIAM J. Comput.*, 10:718–720, 1981.
- [7] C. Ortiz, N. Maculan e J. L. Szwarcfiter. Characterizing and edge-colouring split-indifference graphs. *Discrete Applied Math.*, 82:209–217, 1998.
- [8] M. Plantholt. The chromatic index of graphs with a spanning star. *Journal of Graph Theory*, 5:45–53, 1981.
- [9] V. G. Vizing. On an estimate of the chromatic class of a p -graph. *Diket. Analiz.*, 3:25–30, 1964.

27. Geração de Dados para o Teste de Conformidade baseado em MEF Estendida

Autores: Thaise Yano, Eliane Martins e Fabiano Luís de Sousa ⁶

No teste de conformidade, a especificação do software é utilizada para estabelecer os requisitos de teste, a fim de verificar se a implementação em teste está conforme sua especificação. Assim, como os dados de teste são derivados a partir da especificação, é importante uma especificação correta, consistente e sem ambigüidades para apoiar sua aplicação. Isso se torna fundamental no teste de sistemas reativos, contexto no qual se insere este trabalho, devido à necessidade de que funcionem com alto grau de confiabilidade e segurança. Os sistemas reativos caracterizam-se por interagir continuamente com o ambiente, reagindo a eventos externos e internos [5]. A especificação desses sistemas requer o relacionamento de entradas e saídas no tempo, envolvendo seqüências complexas de eventos, ações, condições e fluxo de informações que se combinam para formar o comportamento do sistema. Dessa forma, o uso de uma especificação formal é importante, não só para o desenvolvimento desse tipo de sistema, mas também para a derivação dos casos de teste e análise dos resultados. Através da sua base matemática, métodos formais fornecem meios de se definir com precisão noções de consistência, correção e completitude. Como exemplos desses métodos podem-se citar a Máquina de Estados Finitos (MEF) e a Máquina de Estados Finitos Estendida (MEFE). Neste trabalho, será adotada a MEFE como modelo de especificação para o teste de conformidade.

Uma MEF [4] é uma máquina hipotética composta por estados e transições. A cada instante, a máquina pode estar em somente um de seus estados. Em resposta a um evento de entrada, a máquina gera um evento de saída e muda de estado. Tanto o evento de saída gerado quanto o novo estado são definidos unicamente em função do estado atual e do evento de entrada. MEF é um modelo simples, mas pode apenas representar a parte de controle de uma especificação. Assim, modelos de MEFE são geralmente utilizados para descrever tanto a parte de controle quanto a de dados. Uma MEFE é uma generalização do modelo clássico da MEF que permite a representação de variáveis, parâmetros de interações e predicados nas transições.

Para gerar os testes baseados em MEF, são realizados percursos no modelo, começando em um nó (geralmente o estado inicial) e terminando em um outro. Métodos tradicionais de teste para MEF não são muito adequados à MEFE, pois podem gerar seqüências de teste não executáveis. Isso acontece devido à existência de predicados nas transições, que não são considerados nos testes baseados em MEF. Como esta última não permite a representação de dados, os testes baseados em MEF também não cobrem o fluxo de dados do modelo. Foram propostos trabalhos que combinam o teste de fluxo de controle e de dados de MEFE [2, 6] e trabalhos que cobrem apenas a parte de dados [10]. Um aspecto importante a ser considerado no

⁶O último autor é membro do Instituto Nacional de Pesquisas Espaciais (INPE)

teste de dados é a seleção de valores apropriados para os parâmetros dos eventos de entrada, que permitam satisfazer aos predicados das condições de guarda de um percurso escolhido no modelo, e que corresponde a um caso de teste.

A geração de dados para satisfazer os requisitos de teste é uma das fases do processo de teste que exige um trabalho intensivo. A automação da geração de caso de teste é um importante passo para reduzir o custo de desenvolvimento de software. Porém, o processo de geração automática de teste tem sido limitado, sendo restringidos pelo tamanho e complexidade do software. Isso motivou diversos pesquisadores aplicarem métodos de otimização para a geração automática dos dados.

A idéia básica da aplicação de técnicas de otimização na geração de dados de teste é buscar dados do domínio de entrada que satisfaçam ao objetivo do teste. Para tanto, o critério de teste pode ser transformado em um problema de otimização que é resolvido com técnicas de busca. O domínio de entrada do software em teste representa o espaço de busca e os dados que satisfazem aos objetivos de testes são as metas. As soluções da busca são comparadas e avaliadas em relação às metas, direcionando a busca para as regiões mais promissoras do espaço de busca.

Como técnica de busca na geração de dados de teste, tem-se aplicado *hill-climbing* [3], *simulated annealing* [9] e algoritmos evolutivos [1, 7, 8]. A aplicação de algoritmos evolutivos no contexto de teste, foco deste trabalho, é conhecida como testes evolutivos. Os algoritmos evolutivos usam a idéia de simulação da evolução como uma estratégia de busca para evoluir as soluções candidatas, usando operadores inspirados pela genética e seleção natural. Em geral, as abordagens usadas até então pelos testes evolutivos visam ao testes baseado no código, os quais buscam entradas que percorram caminhos de execução do programa, utilizando o grafo de fluxo de controle derivado a partir do código do programa. Segundo McMinn et al [7], existem alguns problemas que devem ser tratados no teste evolutivo. Há o problema de estado que está relacionado ao armazenamento de informações em variáveis internas, que retêm valores de uma execução da função em teste para a próxima. Tais variáveis são ocultadas do processo de otimização pois não são disponíveis para a manipulação externa. Assim, faltam informações para o algoritmo decidir se a busca está melhorando ou não, uma vez que o estado geralmente não é variável de entrada, mas influencia na execução. Além disso, os eventos de entrada devem ser fornecidos em uma ordem específica, para colocar a implementação no estado desejado para que um determinado requisito de teste seja atingido. Neste trabalho pretende-se explorar os algoritmos evolutivos na geração de dados para o teste de conformidade, ou seja, para testes baseados em modelos. Como em MEFÉ existe um conjunto de variáveis, o problema de ocultação de variáveis no processo de otimização é um aspecto que deve ser abordado, assim como a seqüência de eventos que devem ser executados para satisfazer predicados associados às transições. Portanto, ao invés de considerar a geração de dados a partir do código, como ocorre nos testes evolutivos, a solução pode estar em gerar os dados a partir da própria MEFÉ.

Em linhas gerais este trabalho visa a estabelecer uma estratégia de geração de dados para o teste de conformidade, utilizando MEFÉ como especificação, e a in-

vestigar a aplicação de métodos de otimização na geração automática de dados de teste. Os principais resultados a serem alcançados com a realização desse trabalho são: (i) estudos teóricos e empíricos de geração automática de dados de teste; (ii) investigação de métodos de otimização para a geração de dados de teste; (iii) estabelecimento de uma estratégia de teste para o teste de conformidade; (iv) especificação e desenvolvimento de uma ferramenta para apoiar a aplicação da estratégia definida.

Referências

- [1] B. T. Abreu. Uma abordagem evolutiva para a geração automática de dados de teste. Master's thesis, Universidade de Campinas, Instituto de Computação, Campinas, SP, July 2006.
- [2] C. Bourhfir, R. Dssouli, and E. M. Aboulhamid. Automatic test generation for efsm-based systems, 1996. Disponível online em <http://citeseer.ist.psu.edu/114451.html>, último acesso em 17/11/2006.
- [3] R. S. Boyer, B. Elspas, and K. N. Levitt. SELECT - a formal system for testing and debugging programs by symbolic execution. In *Proceedings of the international conference on Reliable software*, pages 234–245, New York, NY, USA, 1975. ACM Press.
- [4] A. Gill. *Introduction to the Theory of Finite-State Machines*. McGraw-Hill, New York, 1962.
- [5] D. Harel. Statecharts: A visual formalism for complex systems. *Science of Computer Programming*, 8:231–274, 1987.
- [6] E. Martins, S. B. Sabião, and A. M. Ambrosio. Condata: a tool for automating specification-based test case generation for communication systems. In *33rd Hawaii International Conference on System Sciences*, Maui, Hawaii, January 2000.
- [7] P. McMinn and M. Holcombe. The state problem for evolutionary testing. In *GECCO - Genetic and Evolutionary Computation Conference*, pages 2488–2497, Chicago, USA, July 2003. Springer-Verlag.
- [8] H. Sthamer. *The Automatic Generation of Software Test Data Using Genetic Algorithms*. PhD thesis, University of Glamorgan, Wales, England, 1996.
- [9] N. Tracey, J. Clark, and K. Mander. Automated program flaw finding using simulated annealing. In *ISSTA '98: Proceedings of the 1998 ACM SIGSOFT international symposium on Software testing and analysis*, pages 73–81, New York, NY, USA, 1998. ACM Press.
- [10] H. Ural and B. Yang. A test sequence selection method for protocol testing. *IEEE Transactions on Communications*, 39(4):514–523, April 1991.

28. Relax-and-Cut algorithms to 0-1 Combinatorial Optimization problems

Authors: Victor Fernandes Cavalcante and Cid Carvalho de Souza

Introduction. The term Relax-and-Cut (R&C) was coined to denote a class of Lagrangian relaxation algorithms where inequalities may only be explicitly dualized when they become violated at a Lagrangian relaxation solution [5, 7]. These algorithms appear as a promising alternative approach to strengthen Lagrangian relaxation bounds. They use a dynamic inequality dualization scheme that renders viable the application of Lagrangian relaxation to formulations described by an exponential number of inequalities. Indeed, attempts to do that for the Traveling Salesman Problem [2] date from the early 1980's.

This work main goal lies in investigating the suitability of Relax-and-Cut algorithms to solve 0-1 combinatorial optimization problems. We focus our investigation on two NP-hard problems: the set partitioning problem (SPP) [1] and the vertex separator problem (VSP) [3]. The Set Partitioning is considered the optimization problem with the most widespread application among all special structures in pure integer programming [1]. Its irrefutable importance to combinatorial optimization make that algorithms for solving it also abound [4]. The VSP was first defined in a recent paper of Balas and de Souza [3], where a polyhedral investigation was conducted. It is a graph problem that has applications to various connectivity problems, being the most important ones associated to the efficient solution of linear systems [8]. To tackle both problems, Lagrangian relaxation frameworks within Relax-and-Cut algorithms are being developed along the current work.

Relax-and-Cut Algorithms. Denote by X a subset of $\mathbb{B}^n = \{0, 1\}^n$ and let

$$Z = \min \{cx : Ax \leq b, x \in X\} \quad (1)$$

be a formulation for a \mathcal{NP} -hard combinatorial optimization problem. In association with (1) one has $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$ and $A \in \mathbb{R}^{m \times n}$, where m and n are positive integral values representing, respectively, the number of constraints and the number of variables involved. Assuming that m is an exponential function of n , let Z' denote the formulation obtained after removing constraints $Ax \leq b$ from (1). Also, assume that Z' can be solved in polynomial or pseudo-polynomial time in the problem size. A Lagrangian relaxation of (1) is obtained by bringing the term $\lambda(Ax - b)$ into the objective function of Z' , where $\lambda \in \mathbb{R}_+^m$ is the corresponding vector of Lagrange multipliers. The resulting *Lagrangian relaxation Problem* (LRP) is

$$Z(\lambda) = \min \{cx + \lambda(Ax - b) : x \in X\} = \min \{(c + \lambda A)x - \lambda b : x \in X\}. \quad (2)$$

It is a known fact that $Z(\lambda) \leq Z$ and, therefore, the tightest possible lower bound on Z , attainable through $LRP(\lambda)$, is given by an optimal solution to the *Lagrangian dual problem* (LDP) $Z_D = \max_{\lambda \in \mathbb{R}_+^m} \{\min \{(c + \lambda A)x - \lambda b : x \in X\}\}$.

Several methods exist to compute the LDP. Among these, due to its simplicity and the acceptable results it returns, the subgradient method (SM) is the most widely used. Clearly, if there are exponentially many inequalities in (1), the use of traditional Lagrangian relaxation becomes impracticable. Alternatively, the R&C scheme proposes a dynamic strategy to dualize inequalities. In the literature two strategies to implement R&C algorithms are discussed. They differ, basically, on the moment at which the new inequalities are dualized. In our implementation, we adopt the Non Delayed Relax-and-Cut strategy [6]. Thus, one identifies inequalities that are violated at \bar{x}^k , an optimal solution to (2) at iteration k of SM. To do so, likewise polyhedral cutting-plane generation, a separation problem must be solved at every iteration of SM. Thus, one tries to find at least one inequality violated by the current LRP solution. The inequalities thus identified are candidates to be dualized. It is worth noting that separation problems arising in R&C algorithms may be easier than their polyhedral cutting-plane algorithm counterparts. That applies since LRP normally has integral valued solutions (cf. [7]).

Current Results and Contributions. Aiming to assess our approach, experimentations are being carried out and preliminary computational results were already obtained. Such results reveal that our implementations are competitive with (or outperform) the best approaches currently available in literature to solve the SPP and the VSP. Supported by these results we can attest that the main contribution of this work is twofold and is related to the different usages we made of the R&C algorithms. First, we consider the performance of employing these algorithms as a preprocessing tool that is combined with an Integer Linear Programming solver. The target here is to generate cuts that are stored into a pool and used to feed an algorithm based on linear programming. Computational experiments demonstrated that such combined approach can improve the performance of the B&B implemented by XPRESS solver when used to solve hard SPP instances [4]. Similarly, we found out that the resulting hybridization is a very powerful technique to tackle hard instances of VSP, outperforming the pure B&C algorithm [8] in most of the benchmark instances. Hence, our approach that combines our R&C with the B&C in [8] can be viewed as the best exact method available so far for the VSP.

Also, the R&C algorithms developed to both problems can be used as very good heuristics, being competitive, in terms of CPU time and quality of bounds, with the best algorithms available in literature. For instance, to the SPP, our heuristic was capable of generating high-quality solutions, with optimal values being attained for 66% of the instances tested. Also, the heuristic developed to the VSP proved to be very effective, rapidly reaching the optimum in most of the cases.

Additionally to what have been done up to now, some investigation is being currently performed with the aim of improving our algorithms. Different known classes of valid inequalities to the VSP and alternative compositions of frameworks based on R&C algorithms are being considered to both problems.

Referências

- [1] E. Balas and M.W. Padberg. *Combinatorial Optimization*. Set Partitioning - A Survey. John Wiley & Sons, 1979. Chichester, NY, Brisbane, Toronto.
- [2] E. Balas and N. Christofides. A restricted lagrangean approach to the traveling salesman problem. *Mathematical Programming*, 21:19–46, 1981.
- [3] E. Balas and C.C. de Souza. The Vertex Separator Problem: a polyhedral investigation. *Mathematical Programming*, 2005. 103:583–608, 2005.
- [4] V. F. Cavalcante, C.C de Souza and A. Lucena. A Relax-and-Cut algorithm to the set partitioning problem. Accepted to *Computers and Operations Research*, 2006.
- [5] A. Lucena. Steiner problem in graphs: Lagrangean relaxation and cutting-planes. In *COAL Bulletin*, volume 21. Mathematical Programming Society, 1992.
- [6] A. Lucena. Non delayed relax-and-cut algorithms. *Annals of Operations Research*, 140:375–410, 2005.
- [7] C. Martinhon, A. Lucena, and N. Maculan. Stronger k-tree relaxations for the vehicle routing problem. *European Journal of Operational Research*, 158(1):56–71, 2004.
- [8] E. Balas and C.C. de Souza. The Vertex Separator Problem: algorithms and computations. *Mathematical Programming*, 2005. 103:609–631, 2005.

29. Algoritmos para Problemas de Empacotamento

Autores: Eduardo C. Xavier e Flávio K. Miyazawa

Como trabalho de doutorado buscamos fazer um estudo na área de otimização combinatória, mais especificamente sobre problemas de empacotamento. Quando nos referimos a problemas de empacotamento, estamos tratando de uma grande classe de problemas onde temos um ou mais objetos grandes n -dimensionais, os quais chamamos de recipientes, e vários objetos menores também n -dimensionais os quais chamamos de itens. O nosso objetivo é empacotar itens dentro de recipientes maximizando ou minimizando uma dada função objetivo. Talvez, os dois problemas de empacotamento mais conhecidos sejam o problema de empacotamento unidimensional (*Bin Packing Problem*) e o problema da mochila (*Knapsack Problem*). No primeiro problema temos uma lista de itens e um número infinito de recipientes iguais. O objetivo é empacotar todos os itens no menor número de recipientes possível. No segundo problema, temos um único recipiente e uma lista de itens, cada item com um determinado valor. O objetivo do problema é empacotar itens da lista que maximizem a soma de seus valores.

Problemas de empacotamento possuem aplicações em diversas áreas da Computação e Pesquisa Operacional. Podemos citar como exemplos, problemas em alocação de recursos em computadores ou problemas clássicos de corte de materiais em indústrias [1–9].

Muitas das variações de problemas de empacotamento são problemas de otimização que pertencem à classe NP -difícil. Problemas de otimização, na sua forma geral, têm como objetivo maximizar ou minimizar uma função definida sobre um certo domínio. A teoria clássica de otimização trata do caso em que o domínio é infinito. Já no caso dos chamados problemas de otimização combinatória, o domínio é tipicamente finito; além disso, em geral é fácil listar os seus elementos e também testar se um dado elemento pertence a esse domínio. Ainda assim, a idéia ingênua de testar todos os elementos deste domínio na busca pelo melhor mostra-se inviável na prática, mesmo para instâncias de tamanho moderado.

Neste trabalho, assumimos a hipótese de que $P \neq NP$. Desta forma, tais problemas de empacotamento que são NP -difíceis, não possuem algoritmos eficientes para resolvê-los otimamente. Muitos destes problemas aparecem em aplicações práticas e há um forte apelo econômico para resolvê-los.

Como não conseguimos resolver tais problemas eficientemente de forma ótima, buscamos alternativas que possam ser úteis. Existem vários métodos que são utilizados na prática como heurísticas, programação inteira, métodos híbridos, redes neurais, algoritmos genéticos, dentre outros.

O foco do nosso trabalho de doutorado está no desenvolvimento de heurísticas para problemas de empacotamento, principalmente aquelas onde conseguimos estabelecer uma razão, no pior caso, entre a solução devolvida pela heurística e a solução ótima. Tais heurísticas são comumente chamadas de algoritmos de aproximação. Neste caso, o algoritmo sacrifica a otimalidade em troca da garantia de

uma solução aproximada computável em tempo polinomial em relação ao tamanho da entrada. Em linhas gerais, algoritmos de aproximação são aqueles que não necessariamente produzem uma solução ótima, mas soluções que estão dentro de um certo fator da solução ótima. Esta garantia deve ser satisfeita para todas as instâncias do problema. Desta forma, devemos dar uma demonstração formal deste fato. Também buscamos o desenvolvimento de heurísticas baseadas no método de geração de colunas. Muitos problemas de empacotamento tem formulações em programação linear cujo número de colunas é muito grande, impedindo que tais programas lineares sejam resolvidos por métodos tradicionais. Muitos destes programas lineares fornecem soluções fracionárias muito próximas das soluções inteiras, então há um grande interesse em resolver tais sistemas lineares, e usa-los para gerar soluções inteiras. Como o número de colunas é muito grande, o método de geração de colunas deve ser aplicado.

Referências

- [1] J. S. Ferreira, M. A. Neves, and P. Fonseca e Castro. A two-phase roll cutting problem. *European J. Operational Research*, 44:185–196, 1990.
- [2] S. Ghandeharizadeh and R. R. Muntz. Design and implementation of scalable continous media servers. *Parallel Computing Journal*, 24(1):91–122, 1998.
- [3] P. Gilmore and R. Gomory. A linear programming approach to the cutting stock problem. *Ops. Res.*, 9:849–859, 1961.
- [4] P. Gilmore and R. Gomory. A linear programming approach to the cutting stock problem - part II. *Ops. Res.*, 11:863–888, 1963.
- [5] P. Gilmore and R. Gomory. Multistage cutting stock problems of two and more dimensions. *Ops. Res.*, 13:94–120, 1965.
- [6] P. Gilmore and R. Gomory. The theory and computation of knapsack functions. *Ops. Res.*, 15:1045–1075, 1967.
- [7] P. S. Wu J. L. Wolf and H. Shachnai. Disk load balancing for video-on-demand-systems. *ACM Multimedia Systems Journal*, 5:358–370, 1997.
- [8] H. Shachnai and T. Tamir. Multiprocessor scheduling with machine allotment and parallelism constraints. *Algorithmica*, 32(4):651–678, 2002.
- [9] H. Shachnai and T. Tamir. Tight bounds for online class-constrained packing. *Theoretical Computer Science*, 321(1):103–123, 2004.

30. Tratamento de Exceções no Desenvolvimento de Sistemas Tolerantes a Falhas Baseados em Componentes

Autores: Fernando Castor Filho e Cecília Mary F. Rubira

Mecanismos de tratamento de exceções [1] foram concebidos como um meio para se gerenciar a complexidade de software tolerante a falhas. Esses mecanismos promovem uma separação textual explícita entre o código “normal”, responsável pelas funcionalidades da aplicação, e o código que torna o sistema tolerante a falhas, a fim de dar suporte à construção de programas mais confiáveis, concisos e fáceis de evoluir. Tratamento de exceções complementa outras abordagens para tornar sistemas mais confiáveis, como transações atômicas [4]. O uso de tratamento de exceções na construção de diversos sistemas reais de grande escala e o fato de diversas linguagens de programação orientadas a objetos modernas, como Java, Ada e C#, e modelos de componentes, como DCOM e EJB, implementarem tratamento de exceções atestam sua importância para a prática atual do desenvolvimento de software.

Frequentemente, uma parte considerável do código de um sistema é dedicada à detecção e ao tratamento de erros. Apesar disso, desenvolvedores costumam focar no comportamento normal das aplicações e só lidar com o código responsável por detectar e tratar erros na etapa de implementação [1,9], de maneira *ad hoc*. Essa prática de projeto e implementação cria uma situação propícia para o aparecimento de falhas de projeto. O código responsável por detectar e tratar erros costuma estar espalhado em diversos módulos de um sistema, misturado ao código responsável pelo comportamento normal, o que dificulta sua estruturação, entendimento e manutenção [8]. Em sistemas baseados em componentes, um fenômeno similar pode ser observado. Há evidência de que desenvolvedores de sistemas baseados na plataforma J2EE, um dos padrões da indústria para DBC, apresentam hábitos de programação, no tocante ao uso de tratamento de exceções, que tornam aplicações vulneráveis a falhas de projeto e difíceis de manter [9]. A isso soma-se o fato de que sistemas baseados em componentes introduzem complicações que não existem na construção de sistemas de software “tradicionais”. Por exemplo, em um sistema baseado em componentes, o código fonte de um ou mais componentes pode não estar disponível no momento em que o sistema é integrado [11].

Este trabalho examina o problema da construção de sistemas tolerantes a falhas baseados em componentes. O enfoque do trabalho está em sistemas de software nos quais a separação entre comportamento normal e comportamento excepcional é feita através de tratamento de exceções. Tendo em vista a ênfase em sistemas baseados em componentes, a investigação se divide em duas partes, uma relativa à integração de componentes de software (*design with reuse*) e outra relativa à sua construção (*design for reuse*), o que resulta em duas perguntas de pesquisa:

1. *Como reduzir o número de erros, em sistemas baseados em componentes, decorrentes de suposições conflitantes ou incompletas sobre o comportamento*

excepcional dos componentes integrados no sistema?

2. *Como melhorar a estruturação interna de componentes de software, de modo a minimizar o impacto do comportamento excepcional desses componentes sobre a sua complexidade total?*

A minha tese propõe soluções para diminuir a quantidade de falhas de projeto introduzidas em sistemas baseados em componentes pelo uso de tratamento de exceções. Para alcançar esse fim, são apresentadas duas abordagens complementares, cada uma centrada em um elemento da dicotomia integração/construção. Ambas têm foco na estrutura do sistema como uma maneira de gerenciar a sua complexidade e reduzir o número de falhas de projeto.

A primeira parte da tese foca na integração de componentes de software para a construção de sistemas confiáveis. Mais especificamente, propomos uma abordagem rigorosa para o projeto arquitetural do comportamento excepcional de sistemas baseados em componentes. Quatro capítulos da tese refinam, explicam e validam a seguinte hipótese, que responde a primeira pergunta de pesquisa:

Hipótese 1: *A descrição rigorosa de como exceções fluem entre componentes no nível arquitetural e a verificação automática de propriedades relativas ao comportamento excepcional do sistema nesse nível de abstração aumentam a chance de que falhas de projeto relacionadas ao comportamento excepcional sejam detectadas antes que o sistema seja implementado.*

A segunda parte da tese trata da construção de componentes de software. Mais especificamente, investigamos a adequação do paradigma de programação orientada a aspectos (AOP) [6] para melhorar a estruturação interna de componentes, através de uma separação explícita entre comportamentos normal e excepcional. Com base nesses fatores, dois capítulos da tese validam e refinam a seguinte hipótese, que responde a segunda pergunta de pesquisa:

Hipótese 2: *O uso de programação orientada a aspectos para modularizar o código de tratamento de exceções de um programa melhora a qualidade desse programa, tanto do ponto de vista do comportamento normal quanto do excepcional.*

As contribuições da tese são as seguintes:

1. Um mecanismo de tratamento de exceções que funciona no nível da arquitetura de software e leva em consideração algumas das particularidades de sistemas baseados em componentes. Esse mecanismo é centrado em um estilo arquitetural específico e evidencia a influência que estilos arquiteturais têm na propagação de exceções.
2. Uma abordagem para dar suporte à descrição e à análise de arquiteturas de software, da perspectiva do seu comportamento excepcional. Essa abordagem leva em consideração a maneira como diferentes estilos arquiteturais influenciam a propagação de exceções e complementa metodologias de desenvolvimento cujo foco é o comportamento excepcional do sistema.

3. Um modelo formal que indica as responsabilidades de cada elemento arquitetural, no que se refere ao lançamento, recebimento e tratamento de exceções, e que permite que certas propriedades úteis relativas ao fluxo de exceções entre esses elementos sejam verificadas de maneira automática.
4. Uma abordagem para a especificação e verificação de sistemas concorrentes cooperativos da perspectiva do seu comportamento excepcional.
5. Uma análise dos fatores que tiveram influência na modularização do código de tratamento de exceções usando aspectos, com base na experiência adquirida através da refatoração de quatro aplicações distintas.
6. Uma avaliação inicial dos efeitos da estruturação de tratamento de exceções usando aspectos quando outros interesses transversais também são modularizados através de aspectos.
7. Um catálogo de cenários de tratamento de exceções que indica, para cada cenário, se é vantajoso usar aspectos para separar comportamento normal e comportamento excepcional.

Referências

- [1] Flaviu Cristian. Exception handling. In T. Anderson, editor, *Dependability of Resilient Computers*, pages 68–97. Blackwell Scientific Publications, 1989.
- [2] Q. Cui and J. Gannon. Data-oriented exception handling. *IEEE Transactions on Software Engineering*, 18(5):393–401, May 1992.
- [3] Rogerio de Lemos and Alexander Romanovsky. Exception handling in the software lifecycle. *International Journal of Computer Science and Engineering*, 16(2):167–181, 2001.
- [4] J. Gray and A. Reuter. *Transaction Processing: Concepts and Techniques*. Morgan Kaufmann, 1993.
- [5] Paulo Asterio C. Guerra et al. A fault-tolerant architecture for component-based software systems. In R. de Lemos, C. Gracek, and A. Romanovsky, editors, *Architecting Dependable Systems*, LNCS 2677. Springer-Verlag, 2003.
- [6] G. Kiczales et al. Aspect-oriented programming. In *Proceedings of the 11th ECOOP*, LNCS 1271, pages 220–242, 1997.
- [7] S. Lacourte. Exceptions in guide, an object-oriented language for distributed applications. In *Proceedings of the 5th ECOOP*, LNCS 512, pages 268–287, July 1991.
- [8] Martin Lippert and Cristina V. Lopes. A study on exception detection and handling using aspect-oriented programming. In *Proceedings of the 22nd ICSE*, pages 418–427, June 2000.
- [9] D. Reimer and H. Srinivasan. Analyzing exception usage in large java applications. In *Proceedings of ECOOP'2003 Workshop on Exception Handling in OO Systems*, July 2003.
- [10] Cecília Mary F. Rubira et al. Exception handling in the development of dependable component-based systems. *Software – Practice and Experience*, 35(5):195–236, March 2005.
- [11] Kurt Wallnau et al. *Building Systems from Commercial Components*. SEI Series in Software Engineering. Addison-Wesley, 2002.

31. Sumarização automática para vídeos de futebol televisivos

Autores: Nielsen Cassiano Simões, Neucimar J. Leite, and Beatriz Marcotegui

A análise automática de vídeo permite realizar uma sumarização definindo um subconjunto relevante para ser reutilizado, segundo suas características. Em geral, esta análise está associada à sua anotação, por meio de marcadores, de alguns atributos como o tipo de vídeo (filme, noticiário, esporte, etc.), gênero (ação, romance, documentário), ou mesmo atributos semânticos relacionados ao conteúdo (presença de um determinado artista, acontecimento de um evento importante), etc. A análise manual dos vídeos pode gerar inconsistências devido à subjetividade do operador ou ao seu cansaço, já que vídeos, em geral, demandam muito tempo para serem processados [5]. Especialmente, em caso de vídeos de esporte, é importante um subconjunto com os principais eventos para serem exibidos em programas esportivos ou noticiários. Além disso, o vídeo sumarizado pode também ser utilizado por um operador para a realização do processo de anotação, reduzindo o tempo gasto neste processo.

Em comparação com outros tipos de vídeos, tais como filmes, noticiários ou comerciais, vídeos de esporte possuem uma característica própria, pela ocorrência de eventos relevantes [1,8]. Em vídeos de tênis, por exemplo, os principais eventos estão relacionados aos segmentos de bola em jogo. Já em vídeos de futebol, destacam-se gols, faltas, aplicações de cartão, pênaltis, escanteios e substituições. Além disso, pode-se identificar para esses tipos de vídeo, padrões de transmissão televisivos, denominados cinemática [4], quando da ocorrência dos eventos.

Um segmento de vídeo digital é composto por uma seqüência de quadros. Um subconjunto ininterrupto de quadros, obtidos de uma mesma câmera, é denominado tomada. Em geral, o processo de análise de vídeo está associado à detecção de transições entre tomadas [3], decorrentes da edição do vídeo. Pode-se também definir alguns quadros para representar cada tomada, denominados quadros-chave. Tais quadros são, normalmente, utilizados para facilitar a navegação no vídeo.

As principais abordagens para anotação automática de vídeos de futebol estão associadas à detecção de eventos específicos, tais como replays [2, 6, 7] e gols [9]. Ekin et. al. [4] fornece uma outra abordagem para a detecção de replays e gols, agregando a detecção de pênaltis e realizando uma sumarização do vídeo baseada em cinemática. Porém, seu modelo requer a interação do operador para a especificação de alguns parâmetros visuais e possui um alto custo computacional.

Este trabalho fornece ferramentas para análise e sumarização de vídeos de futebol transmitidos pela televisão, baseadas na seleção de quadros-chave considerando especialmente o padrão de transmissão da televisão brasileira. Suas principais contribuições são: definição de um modelo para a detecção de eventos em vídeos de futebol; desenvolvimento de ferramentas para a análise de imagens de vídeo, permitindo uma classificação de tomadas; definição de um modelo para seleção de quadro-chave; e

desenvolvimento de ferramentas para análise audio-visual para a detecção de eventos em vídeos de futebol televisivos, permitindo sua sumarização.

Referências

- [1] N. Adami, R. Leonardi, and P. Migliorati. An overview of multi-modal techniques for the characterization of sport programmes. In *SPIE Visual Communications and Image Processing*, volume 5150, pages 1296–1306, July 2003.
- [2] Noboru Babaguchi, Yoshihiko Kawai, Yukinobu Yasugi, and Tadahiro Kitahashi. Linking live and replay scenes in broadcasted sports video. In *MULTIMEDIA '00: Proceedings of the 2000 ACM workshops on Multimedia*, pages 205–208. ACM Press, 2000.
- [3] R. Brunelli, O. Mich, and C. Modena. A survey on video indexing. Technical Report 9612-06, Instituto per la Ricerca Scientifica e Tecnologica, 1996.
- [4] Ahmet Ekin, A.Murat Tekalp, and Rajiv Mehrotra. Automatic soccer video analysis and summarization. *IEEE Transactions on Image Processing*, 12(7):796–807, July 2003.
- [5] E. Kijak, L. Oisel, and P. Gros. Hierarchical structure analysis of sport videos using HMMS. In *International Conference on Image Processing*, volume 2, pages II–1025–II–1028, September 2003.
- [6] Vikrant Kobla, Daniel DeMenthon, and David Doermann. Identifying sports videos using replay, text and camera motion features. In *SPIE Conference on Storage and Retrieval for Image and Video*, pages 332–343, January 2000.
- [7] Hao Pan, P. van Beek, and M. Ibrahim Sezan. Detection of slow-motion replay segments in sports video for highlights generation. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 3, pages 1649–1652, 2001.
- [8] Jinjun Wang, Changsheng Xu, Engsiong Chng, and Qi Tian. Sports highlight detection from keyword sequences using HMM. In *IEEE International Conference on Multimedia and Expo (ICME)*, volume 1, pages 599–602, June 2004.
- [9] Ziyong Xiong, Regunathan Radhakrishnan, and Ajay Divakaran. Generation of sports highlights using motion activity in combination with a common audio feature extraction framework. In *International Conference on Image Processing*, volume 1, pages I–5–I–8, September 2003.

32. Reconhecimento de Parasitos Intestinais do Homem em Imagens de Microscopia Ótica

Autores: João Paulo Papa e Alexandre Xavier Falcão

O reconhecimento de padrões é um processo de descobrimento, identificação e entendimento de padrões que sejam relevantes à performance de uma tarefa, sejam elas oriundas de quaisquer fontes de dados. Uma das principais metas do reconhecimento é possibilitar a uma máquina a capacidade de aproximar-se, em um determinado sentido, à potencialidade do sistema visual humano, reconhecendo-se, dessa forma, a complexidade de tal tarefa [1]. Neste trabalho, estamos interessados no reconhecimento de padrões através da análise de imagens.

Devido à heterogeneidade e complexidade das aplicações que envolvem um sistema de reconhecimento de padrões, a divisão do mesmo em três áreas básicas torna-se, portanto, conceitualmente útil, sendo elas: segmentação, extração de características e classificação [2].

A segmentação de imagens, por sua vez, refere-se à decomposição da imagem em partes constituintes (objetos) relevantes para o domínio da aplicação. Um exemplo é a imagem de um automóvel fotografado por um radar, por atingir o limite máximo de velocidade. Técnicas de segmentação podem ser aplicadas à imagem, de forma a extrair áreas de interesse da mesma, tais como a placa traseira que permite a identificação do veículo, e conseqüentemente sua origem e seu proprietário. Nota-se, dessa forma, a extrema importância da segmentação de imagens para as etapas subseqüentes: extração de características e classificação de padrões [3], as quais são o foco de estudo do presente projeto. Uma vez que a imagem está segmentada em objetos, características de forma, cor e textura desses objetos são extraídas para gerar descritores para classificação. A classificação associa os objetos às classes que eles pertencem com base na similaridade entre seus descritores.

Uma grande área, a qual tem crescido muito nos últimos anos, está relacionada a aplicações de análise de imagens médicas como, por exemplo, diagnósticos tumorais, mensurações de órgãos, contagem e identificação de células e detecção, prevenção e controle de doenças. É nessa última aplicação que está concentrado o presente projeto de pesquisa, ou seja, desenvolver um sistema de reconhecimento automático de padrões para diagnóstico de parasitos intestinais em humanos. É necessário, também, salientar que a etapa de segmentação será desenvolvida por um aluno de mestrado em ciência da computação do mesmo instituto de pesquisa, com o devido acompanhamento dos envolvidos no presente projeto.

Atualmente, as parasitoses intestinais no homem ainda são prevalentes no mundo, principalmente em regiões tropicais [4]. No Brasil, as atuais prevalências de infecções provocadas por protozoários e helmintos intestinais justificam o diagnóstico preciso do agente, a fim de introduzir medidas adequadas e, principalmente, rápidas e eficazes para solucionar este problema de saúde pública.

O diagnóstico laboratorial de parasitos intestinais é efetuado por microscopia

ótica, através da identificação visual das formas parasitárias existentes em amostras fecais do homem. Tais amostras fecais provenientes do paciente são previamente submetidas a uma série de procedimentos técnicos no laboratório para remoção do excesso de impurezas. Logo após, uma pequena quantidade da amostra fecal é extraída e colocada em uma lâmina microscópica. Porém, esse procedimento apresenta algumas desvantagens, tais como alto teor de resíduos nas lâminas, pequena quantidade de material que é resgatado para leitura microscópica e a necessidade de um profissional de microscopia que saiba diagnosticar as enteroparasitoses intestinais. Profissional esse que é considerado uma mão-de-obra cada vez mais rara. É neste momento que entra o diagnóstico por computador, utilizando técnicas de reconhecimento de padrões como alternativa para resolver o problema do diagnóstico parasitológico.

Portanto, o presente projeto de pesquisa propõe, como objetivo principal, a automação desse processo de identificação de parasitos intestinais, visando melhorias em rapidez, confiabilidade e custos.

A análise por computador requer a construção de uma base de imagens selecionada por um especialista em parasitologia, contendo várias amostras de cada parasito. Técnicas de reconhecimento de padrões serão investigadas [5] com o objetivo de criar um algoritmo capaz de extrair características relevantes, tais como informações morfológicas e topológicas dos parasitos, e identificar as classes que eles pertencem utilizando esses dados. A essas informações podemos relacionar a forma do contorno externo da membrana e de outras subestruturas, tais como os núcleos, bem como o número dos mesmos.

Em sua primeira versão, o diagnóstico por computador funcionará como uma segunda opinião para o microscopista, refletindo com um certo grau de confiabilidade o conhecimento do especialista que gerou a base de imagens.

Aliado ao projeto está o desenvolvimento de um novo classificador de padrões, bem como estudo e desenvolvimento de técnicas de combinação de classificadores e descritores.

Referências

- [1] R. C. Gonzalez and R. E. Woods. *Processamento Digital de Imagens*. Edgard Blücher, 2000.
- [2] A. K. Jain. *Fundamentals of Digital Image Processing*. Prentice Hall, 1989.
- [3] K. R. Castleman. *Digital Image Processing*. Prentice Hall, 1996.
- [4] L. Rey. *Bases da Parasitologia Médica*. Guanabara Koogan, 2002.
- [5] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*. John Wiley & Sons, 1973.