UFMG/ICEx/DCC Projeto e Análise de Algoritmos

Trabalho Prático da Disciplina

Programa de Pós-Graduação em Ciência da Computação 1º Semestre de 2007

1 Introdução

O ensino de Pós-Graduação (Mestrado e Doutorado) e a pesquisa devem andar juntos. Na verdade, em instituições onde as duas atividades estão presentes, cada uma delas deve influenciar a outra positivamente. Esse é o caso da UFMG e, em particular, do Programa de Pós-Graduação em Ciência da Computação, que é fortemente influenciado pelas atividades de pesquisa desenvolvidas pelos membros de seu corpo docente, que acabam influenciando as atividades de ensino.

O objetivo deste trabalho prático é "expor" o aluno a um projeto de pesquisa que deve ter as seguintes características:

- Tratar de um problema que envolva o projeto e análise de algoritmos. No entanto, o trabalho a ser feito deve tratar obrigatoriamente de um problema novo ou de um problema conhecido mas que tenha um enfoque novo. Esse é um critério que definitivamente deve ser observado;
- Fazer a implementação de algoritmos. Espera-se que cada aluno implemente, ao final deste projeto, em torno de 2500 linhas de código C, ou equivalente nas linguagens C++, Java ou Pascal;
- Escrever um artigo técnico de até 15 páginas relatando os resultados do problema no formato definido pela ACM. Para informações sobre a formatação veja a página `http://www.acm.org/sigs/pubs/proceed/template.html`. Esse artigo técnico deve ter basicamente três partes (isto não quer dizer que ele tenha apenas três seções): introdução, desenvolvimento e conclusões. A introdução deve incluir a motivação para estudar o problema, o contexto do problema, o enunciado do problema, um sumário do que foi feito incluindo os resultados obtidos e a organização do artigo. O desenvolvimento deve incluir uma discussão objetiva dos trabalhos relacionados, e toda uma apresentação do que foi feito incluindo os resultados e as avaliações. As conclusões devem apresentar uma visão geral após o trabalho ter sido feito e com sugestões de trabalho futuro. Esse artigo deve ser entregue ao professor antes da apresentação do seminário em data a ser definida.
- Apresentar o trabalho na forma de um seminário. Essa apresentação irá ocorrer em data a ser definida. No entanto, não será depois do dia 6/7/2007.

Este trabalho pode ser feito em grupo. Sugere-se que cada grupo tenha no máximo quatro ou cinco alunos. O ideal é ter grupos de três alunos. Veja que o tamanho final da implementação deve ser proporcional ao número de alunos do grupo.

2 Proposta

Antes de começar a fazer o seu trabalho, entregue em sala de aula até o dia 15/3/2007 (quinta-feira) às 14:50 ou envie uma mensagem eletrônica para `esub.para.loureiro@gmail.com` até esse dia e horário com `Assunto: [PAA] Proposta do Trabalho Prático`, a proposta do que pretende fazer incluindo:

- Descrição do problema justificando a primeira característica descrita acima;
- Referências bibliográficas já consultadas ou a serem consultadas;
- Proposta de cronograma com as atividades previstas para serem executadas. Esta disciplina terá um monitor, que juntamente com o professor, irão fazer um acompanhamento do trabalho através de reuniões a serem marcadas com os membros do grupo fora do horário da aula;

- Membros do grupo.

Caso envie uma mensagem, escreva a proposta no formato texto ou no formato pdf.

Importante: Você nunca terá tanto tempo para fazer este trabalho como agora. Comece a investir nele imediatamente.

3 Sugestões de Temas

A seguir, são apresentadas algumas sugestões de temas que refletem o espírito do trabalho a ser feito.

3.1 Segurança na Transmissão de Dados

Chamada. Veja a seguinte chamada sobre segurança em tempo real e proteção de *copyright* de dados multimídia.

Call for Papers for Special Issue: Circuits and Systems for Real-Time Security and Copyright Protection of Multimedia International Journal of Computers and Electrical Engineering (Elsevier Ltd.)

`(http://www.elsevier.com/wps/find/journaldescription.cws_home/367/description)`

Guest Editor(s):
– Saraju P. Mohanty, University of North Texas, email: `smohanty@cse.unt.edu`
– Nasir Memon, Polytechnic University, email: `memon@poly.edu`
– Karam S. Chatha, Arizona State University, email: `kchatha@asu.edu`

Following the explosive growth of the Internet, concerns about protection and enforcement of IP (Intellectual Property) rights of digital content involved in transactions have been mounting. Easy unauthorized replication and manipulation with inexpensive tools has worsened the scenario. Due to this, the movie/audio industry loses several billions of dollars every year. Hence, DRM (Digital Rights Management) systems are necessary for establishing ownership rights, tracking usage, ensuring authorized access, preventing illegal replication, and facilitating content authentication. Such DRM systems may need various techniques including, encryption, watermarking, steganography, scrambling, digital certificates, secure communications protocols, etc. In the last decade, significant research progress has been made to develop these techniques which primarily work offline. However, in case of emerging applications, such as, digital television broadcasting, internet protocol television (IP-TV), video on demand, pay-TV, electronic passport (e-passport), credit cards, driving licenses, etc. security and copyright protection mechanisms have to work in real-time. Consequently, appliances, like digital cameras, network processors, mobile/video phones, graphics processing units, DVD player, etc. need to be equipped with such mechanisms. In these situations, software only solutions may not be adequate to provide real-time performance, rather hardware assisted solutions is needed for easy integration with multimedia hardware, low-power consumption, higher reliability/availability, and low-cost compared.

The guest editors invite original manuscripts addressing the above challenges. The topics of research interest include, but not limited to:
– Real-time performance integrated circuits for DRM techniques.
– Operating-system and (micro) architecture-level support for security and copyright protection.
– Methods to integrate security and copyright protection mechanism in embedded architectures.
– Building SoCs such as camera, mobile phones, network processors with DRM technology.
– Building media processors, graphics processing units with DRM technology.
– Techniques for secure multimedia broadcasting in wireless systems.
– Security and copyright techniques for home entertainment, such as IP-TV and digital TV, etc.
– Techniques for real-time multimedia processing in encryption and/or watermarking domain.
– Design of side-channel-resistant embedded systems to enable attack-proof DRM development.
– Low-power DRM technology for portable appliances.

Submission Information: Manuscripts should be emailed to the guest editors (`smohanty@cse.unt.edu` and `memon@poly.edu` and `kchatha@asu.edu`) as pdf files. The maximum page limit for a manuscript is 25 pages. The pdf file should be named using last name of the first author followed by a keyword from the paper title. For example, the filename for authors S. Mohanty and N. Memon submitting the paper titled: "On Watermarking Technique's Developments" would be Mohanty-Watermarking. Guidelines for manuscript preparation can be found at: `http://www.elsevier.com/wps/find/journaldescription.cws_home/367/authorinstructions.`

Schedule:
– Paper Submission Deadline: 15th June 2007
– Notification of Acceptance or Request for Revision (if any): 15th August 2007
– Notification of Revised Papers Acceptance: 31st August 2007
– Camera Ready Final Version Due: 30th September 2007
– Appearance of Special Issue: December 2007/Early 2008

Questions and More Information: Please feel free to contact the guest editors.

Referências. A seguir, estão algumas definições das técnicas descritas acima, que foram retiradas e adaptadas da Enciclopédia Wikipedia em inglês (`en.wikipedia.org`).

In **cryptography**, **encryption** is the process of obscuring information to make it unreadable without special knowledge, sometimes referred to as scrambling. Encryption has been used to protect communications for centuries, but only organizations and individuals with an extraordinary need for secrecy had made use of it. In the mid-1970s, strong encryption emerged from the sole preserve of secretive government agencies into the public domain, and is now used in protecting widely-used systems, such as Internet e-commerce, mobile telephone networks and bank automatic teller machines.

Encryption can be used to ensure secrecy, but other techniques are still needed to make communications secure, particularly to verify the integrity and authenticity of a message; for example, a **Message Authentication Code (MAC)** or **digital signatures**. Another consideration is protection against traffic analysis.

Encryption or software code obfuscation is also used in software copy protection against reverse engineering, unauthorized application analysis, cracks and software piracy used in different encryption or obfuscating software.

In cryptography, a **cipher** (or cypher) is an algorithm for performing encryption and decryption – a series of well-defined steps that can be followed as a procedure. An alternative term is encipherment. In most cases, that procedure is varied depending on a key which changes the detailed operation of the algorithm. In non-technical usage, a "cipher" is the same thing as a "code"; however, the concepts are distinct in cryptography. In classical cryptography, ciphers were distinguished from codes, which operated by substituting according to a large codebook.

The original information is known as plaintext, and the encrypted form as ciphertext. The ciphertext message contains all the information of the plaintext message, but is not in a format readable by a human or computer without the proper mechanism to decrypt it; it should resemble random gibberish to those not intended to read it.

Most modern ciphers can be categorized in several ways:
– By whether they work on blocks of symbols usually of a fixed size (block ciphers), or on a continuous stream of symbols (stream ciphers);
– By whether the same key is used for both encryption and decryption (symmetric key algorithms), or if a different key is used for each (asymmetric key algorithms). If the algorithm is symmetric, the key must be known to the recipient and to no one else. If the algorithm is an asymmetric one, the encyphering key is different from, but closely related to, the decyphering key. If one key cannot be deduced from the other, the asymmetric key algorithm has the public/private key property and one of the keys may be made public without loss of confidentiality. The Feistel cipher uses a combination of substitution and transposition techniques. Most (block ciphers) algorithms are based on this structure.

A security protocol (cryptographic protocol or encryption protocol) is an abstract or concrete protocol that performs a security-related function and applies cryptographic methods.

A protocol describes how the algorithms should be used. A sufficiently detailed protocol includes details about data structures and representations, at which point it can be used to implement multiple, interoperable versions of a program.

Cryptographic protocols are widely used for secure application-level data transport. A cryptographic protocol usually incorporates at least some of these aspects:
– Key agreement or establishment;
– Entity authentication;
– Symmetric encryption and message authentication material construction;
– Secured application-level data transport;
– Non-repudiation methods.

Steganography is the art and science of writing hidden messages in such a way that no one apart from the intended recipient knows of the existence of the message; this is in contrast to cryptography, where the existence of the message itself is not disguised, but the content is obscured.

Generally, a steganographic message will appear to be something else: a picture, an article, a shopping list, or some other message. This apparent message is the covertext. For instance, a message may be hidden by using invisible ink between the visible lines of innocuous documents.

The advantage of steganography over cryptography alone is that messages do not attract attention to themselves, to messengers, or to recipients. An unhidden coded message, no matter how unbreakable it is, will arouse suspicion and may in itself be incriminating, as in some countries encryption is illegal.

A steganographic message (the plaintext) is often first encrypted by some traditional means, and then a covertext is modified in some way to contain the encrypted message (ciphertext), resulting in stegotext. For example, the letter size, spacing, typeface, or other characteristics of a covertext can be manipulated to carry the hidden message; only the recipient (who must know the technique used) can recover the message and then decrypt it.

Watermarking imprints the image with a watermark that is not visible to the naked eye, but can be detected by another piece of software. The detection software can be embedded within a spider that crawls all over the network (e.g., World Wide Web) and finds all the watermarked data (e.g., image, audio), making sure that no one is treating it (e.g., displaying).

Cryptography researchers have developed different techniques to remove watermarks from different media. The challenge is to develop an algorithm that produces watermarks that survive media manipulation techniques (e.g., photo editing) and cryptographic attacks designed to remove watermarks.

O que pode ser feito. Os problemas que podem ser resolvidos, a partir da chamada de trabalhos acima, estão representados, de forma genérica, na figura 1.

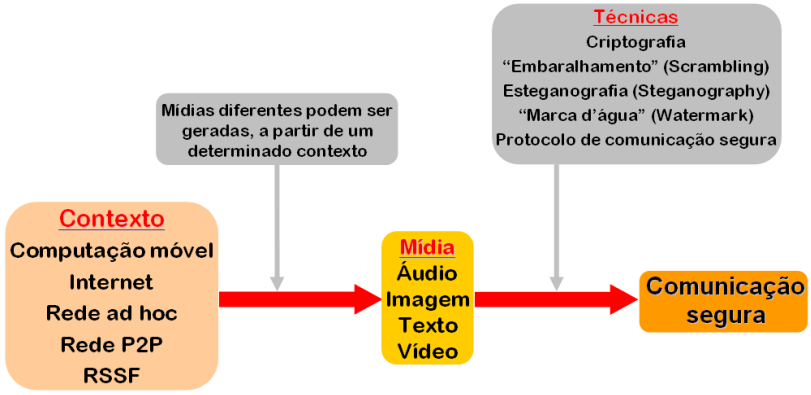

Figura 1: Problema de segurança na transmissão de dados

Como discutido na chamada de trabalhos, este problema pode ser estudado em diferentes contextos tratando de diferentes tipos de mídia e usando diferentes tipos de técnicas. Veja que se for considerado um contexto, um tipo de mídia e uma técnica, existem (potencialmente) 80 possibilidades diferentes, como mostrado na equação 1, não considerando nenhum outro critério no problema como algum parâmetro de QoS.

$$\begin{array}{ccccc} \textbf{Contexto} & & \textbf{Mídia} & & \textbf{Técnica} \\ \begin{pmatrix} \text{Comp Móvel} \\ \text{Internet} \\ \text{Rede ad hoc} \\ \text{Rede P2P} \\ \text{RSSF} \end{pmatrix} & \times & \begin{pmatrix} \text{Áudio} \\ \text{Imagem} \\ \text{Texto} \\ \text{Vídeo} \end{pmatrix} & \times & \begin{pmatrix} \text{Embaralhamento} \\ \text{Esteganografia} \\ \text{Marca d'água} \\ \text{Prot comunicação segura} \end{pmatrix} \end{array} = 80 \qquad (1)$$

3.2 Aritmética Computacional

Chamada. Veja a seguinte chamada sobre aritmética computacional.

Call for Papers for Special Issue on Computer Arithmetic of the IEEE Transactions on Computers, September Issue of 2008

(http://www.computer.org/portal/cms_docs_transactions/transactions/tc/CFP/t0144.pdf)

IEEE Transactions on Computers seeks original manuscripts for a Special Section on Computer Arithmetic scheduled to appear in the September issue of 2008. Computer arithmetic is fundamental to the design of general-purpose and application-specific processors. Novel arithmetic algorithms and hardware designs are needed to satisfy the requirements of numerically-intensive applications in a variety of areas including scientific computing, cryptography, multimedia, graphics and digital signal processing. Specialized number representations and encodings of these play a significant role in the design of efficient arithmetic algorithms and their implementations. Thus also understanding fundamental properties of finite precision number systems is essential in the engineering of efficient arithmetic algorithms, as well as the available and emerging technologies are important for the design, choice and implementation of such algorithms.

Topics of interest are recent advances in all aspects of computer arithmetic, including, but not limited to:

1. Arithmetic processor design and implementation;
2. Arithmetic algorithms and their analysis;
3. High-performance arithmetic units and systems;
4. New floating-point units and systems;
5. Foundations of number systems and arithmetic;
6. Implementations of arithmetic units using reconfigurable logic;
7. Verification, testing and error analysis for computer arithmetic;
8. Standards for number representations and arithmetic;
9. Elementary and special function evaluation;
10. Decimal arithmetic;
11. Low power arithmetic; and
12. Applications of computer arithmetic in cryptography, DSP, multimedia, data compression, graphics, etc.

Submitted articles must not have been previously published or currently submitted for journal publication elsewhere. As an author, you are responsible for understanding and adhering to our submission guidelines. You can access them by clicking on `http://www.computer.org/mc/tc/author.htm`. Please thoroughly read these before submitting your manuscript.

Please submit your paper to Manuscript Central at `http://cs-ieee.manuscriptcentral.com/`. Please feel free to contact the Peer Review Supervisor, Suzanne Werner at `swerner@computer.org` or the guest editors at `kornerup@imada.sdu.dk`, `paolo.montuschi@polito.it`, `Jean-Michel.Muller@ens-lyon.fr`, `eschwarz@us.ibm.com` if you have any questions. Please note the following important dates.

Important dates:
– Submission Deadline: July 22, 2007
– Reviews Completed: November 30, 2007
– Major Revisions Due (if Needed): January 13, 2008
– Reviews of Revisions Completed (if Needed): March 14, 2008
– Minor Revisions Due (if Needed): April 6, 2008
– Notification of Final Acceptance: April 22, 2008
– Publication Materials for Final Manuscripts Due: April 30, 2008

Please address all other correspondence regarding this special issue to Guest Editors P. Kornerup, P. Montuschi, J.- M. Muller, E. Schwarz.

GUEST EDITORS
– Peter Kornerup, Univ. of Southern Denmark, Denmark, `kornerup@imada.sdu.dk`
– Paolo Montuschi, Politecnico di Torino, Italy, `paolo.montuschi@polito.it`
– Jean-Michel Muller, LIP/Arénaire ENS Lyon, France, `Jean-Michel.Muller@ens-lyon.fr`
– Eric Schwarz, IBM, USA, `eschwarz@us.ibm.com`

O que pode ser feito. Observe que os itens 2, 9, 10, 11 e 12, descritos acima, tratam do projeto e análise de algoritmos. A "revista" (*journal*) IEEE Transactions on Computers é uma das publicações mais conceituadas da área de Ciência da Computação. Da mesma forma que a sugestão anterior, pode-se estudar este problema

num dado contexto como mostrado na figura 1. Em particular, um contexto bastante promissor é o de Rede de Sensores Sem Fio devido às restrições computacionais dos nós sensores. Ou seja, esta é uma sugestão que permite desenvolver trabalhos bastante interessantes e inovadores.

3.3 Avaliação Experimental de Algoritmos

Chamada. Veja a chamada de trabalhos que define o escopo de publicações do periódico *ACM Journal of Experimental Algorithmics* (`http://www.jea.acm.org/`) e do *Workshop on Algorithms and Data Structures* (`http://www.wads.org`). São dois veículos diferentes para publicação de trabalhos técnicos mas que têm contextos similares.

JEA Call for Papers

`(http://www.jea.acm.org/call.html)`

The ACM JEA is a high-quality, refereed, archival journal devoted to the study of discrete algorithms and data structures through a combination of experimentation and classical analysis and design techniques. ACM JEA is an entirely online journal.

Original submissions are sought that address implementation and performance issues of discrete algorithms and data structures. An experimental study typically includes an implementation, a series of experiments designed to understand the behavior of the algorithm(s) under study, and a critical discussion of the experiments and their results. Whenever possible, experiments should include test data from previously published studies to enable critical comparisons, although the development of new test suites is also encouraged. Studies of an algorithm in a specific application context of general interest are welcome, as are contributions in the development and understanding of experimental methodologies, including multimedia tools such as algorithm animation.

Also within the scope of the ACM JEA are research contributions in the area of test generation and result assessment as applied to discrete algorithms and data structures. Fundamental and application areas include, but are not limited to: combinatorial optimization, computational biology, computational geometry, graph manipulation, graphics, heuristics, network design, parallel processing, routing, searching and sorting, scheduling, and VLSI design.

Submissions to JEA typically include an article, a suite of programs, and a collection of test data and computational results. Accepted submissions are placed on-line, with code and data made available for use by researchers and practitioners alike.

Submissions, refereeing, and all correspondence are conducted through the Internet. For detailed information, visit the site `jea.acm.org` or send an inquiry to `editor@jea.acm.org`.

Call For Papers, Workshop on Algorithms and Data Structures (WADS), Aug 15 – Aug 17, 2007, Halifax, Canada

`(http://www.wads.org)`

The Workshop, which alternates with the Scandinavian Workshop on Algorithm Theory, is intended as a forum for researchers in the area of design and analysis of algorithms and data structures. We invite submissions of papers presenting original research on the theory and application of algorithms and data structures in all areas, including combinatorics, computational geometry, databases, graphics, parallel and distributed computing.

Contributors are invited to submit a full paper (not exceeding 12 pages). Detailed submission instructions are located at `http://www.wads.org`. Submissions must arrive on or before Feb 23, 2007 at 11:59 pm (midnight) EST. Authors will be notified of acceptance or rejection by April 23, 2007. Proceedings will be published in the Springer Verlag series Lecture Notes in Computer Science. The final versions of accepted papers must arrive in camera-ready form before May 11, 2007 to ensure the availability of the proceedings at the conference. Selected papers will be invited for publication in a special issue of Computational Geometry: Theory and Applications.

INVITED SPEAKERS:
– Jeff Erickson (University of Illinois at Urbana-Champaign, USA)
– Mike Langston (University of Tenessee, USA)

CONFERENCE CHAIR AND LOCAL ARRANGEMENTS CHAIR: Norbert Zeh (Dalhousie)

PROGRAM COMMITTEE

Co-Chairs: Frank Dehne (Carleton), Joerg-Rudiger Sack (Carleton), Norbert Zeh (Dalhousie).

PC-Members:
Susanne Albers (University of Freiburg, Germany)
Alberto Apostolico (University of Padova, Italy)
Tetsuo Asano (JAIST, Japan)
Mike Atallah (Purdue University, USA)
Mark de Berg (University of Eindhoven, Netherlands)
Gerth Brodal (University of Aarhus, Denmark)
Timothy Chan (University of Waterloo, Canada)
Danny Ziyi Chen (University of Notre Dame, USA)
Tom Cormen (Dartmouth College, USA)
Camil Demetrescu (University of Rome "La Sapienza", Italy)
David Eppstein (University of California, Irvine, USA)
Susanne Hambrusch (Purdue University, USA)
Rolf Klein (University of Bonn, Germany)
Mike Langston (University of Tennessee, USA)
Andrzej Lingas (Lund University, Sweden)
Joe Mitchell (SUNY Stonybrook, USA)
David Mount (University of Maryland, USA)
Andrew Rau-Chaplin (Dalhousie University, Canada)
Arnold Rosenberg (University of Massachusetts, USA)
Roberto Tamassia (Brown University, USA)

O que pode ser feito. Existem vários algoritmos que podem ser avaliados, de acordo com o contexto do JEA ou do workshop WADS (neste ano, as submissões para o WADS '07 foram encerradas em fevereiro). Veja, por exemplo, o artigo intitulado *Simultaneous Optimization for Concave Costs: Single Sink Aggregation or Single Source Buy-at-Bulk* [1] publicado no *journal* **Algorithmica**. A seguir, o resumo desse artigo.

> ***Abstract***. *We consider the problem of finding efficient trees to send information from k sources to a single sink in a network where information can be aggregated at intermediate nodes in the tree. Specifically, we assume that if information from j sources is traveling over a link, the total information that needs to be transmitted is $f(j)$. One natural and important (though not necessarily comprehensive) class of functions is those which are concave, non-decreasing, and satisfy $f(0) = 0$. Our goal is to find a tree which is a good approximation simultaneously to the optimum trees for all such functions. This problem is motivated by aggregation in sensor networks, as well as by buy-at-bulk network design.*
>
> *We present a randomized tree construction algorithm that guarantees $E[max_f C_f/C^*(f)] \leq 1 + \log k$, where C_f is a random variable denoting the cost of the tree for function f and $C^*(f)$ is the cost of the optimum tree for function f. To the best of our knowledge, this is the first result regarding simultaneous optimization for concave costs. We also show how to derandomize this result to obtain a deterministic algorithm that guarantees $max_f C_f/C^*(f) = O(\log k)$. Both these results are much stronger than merely obtaining a guarantee on max_f $E[C_f/C^*(f)]$. A guarantee on max_f $E[C_f/C^*(f)]$ can be obtained using existing techniques, but this does not capture simultaneous optimization since no one tree is guaranteed to be a good approximation for all f simultaneously.*
>
> *While our analysis is quite involved, the algorithm itself is very simple and may well find practical use. We also hope that our techniques will prove useful for other problems where one needs simultaneous optimization for concave costs.*

No artigo publicado, não há uma avaliação experimental do algoritmo. Também nesse trabalho, existem algumas sugestões de extensões que podem ser consideradas. Assim, uma possível sugestão é avaliar esse algoritmo e/ou propor e avaliar alguma extensão descrita no artigo. Ou seja, identifique um artigo técnico publicado recentemente na literatura, que não está avaliado ainda, e faça sua avaliação de acordo com as expectativas desses dois veículos.

3.4 Algoritmos Combinatórios

O problema. No momento, Donald Knuth está preparando o capítulo 7 do volume 4 da série **The Art of Computer Programming** que trata de algoritmos combinatórios. Em dezembro/2006 e janeiro/2007 ele disponibilizou os três fascículos iniciais desse capítulo (`http://www-cs-faculty.stanford.edu/~knuth/news.html`), que tratam dos seguintes assuntos:

- Pre-Fascicle 0b: Boolean Basics (version of 20 Dec 2006) (Section 7.1.1);
- Pre-Fascicle 0c: Boolean Evaluation (version of 20 Dec 2006) (Section 7.1.2);
- Pre-Fascicle 1a: Bitwise Tricks and Techniques (version of 03 Jan 2007) (Section 7.1.3).

O que pode ser feito. O fascículo sobre *bitwise tricks and techniques* descreve, em particular, vários algoritmos que podem ser avaliados em contextos diferentes como descrito na figura 1.

Na literatura técnica de projeto e análise de algoritmos, existem vários outros livros e periódicos que tratam de algoritmos combinatórios. Nesta sugestão, o objetivo é avaliar algoritmos combinatórios para um dado contexto.

3.5 Algoritmos Randômicos ou Probabilísticos

O problema. "A randomized algorithm or probabilistic algorithm is an algorithm which employs a degree of randomness as part of its logic. In common practice, this means that the machine implementing the algorithm has access to a pseudo-random number generator. The algorithm typically uses the random bits as an auxiliary input to guide its behavior, in the hope of achieving good performance in the 'average case'. Formally, the algorithm's performance will be a random variable determined by the random bits, with (hopefully) good expected value; this expected value is called the expected runtime. The 'worst case' is typically so unlikely to occur that it can be ignored". Fonte: Wikipedia (`http://en.wikipedia.org/wiki/Randomized_algorithm`).

O que pode ser feito. Uma sugestão interessante é o estudo de algoritmos randômicos ou probabilísticos que tendem a ter um bom desempenho no caso médio. Em [1], os autores apresentam um algoritmo randômico para construção de uma árvore.

Nesta sugestão, o objetivo é avaliar algoritmos randômicos para um dado contexto. Também pode-se pensar numa versão randômica de um algoritmo determinístico.

Referências

[1] Ashish Goel and Deborah Estrin. Simultaneous optimization for concave costs: Single sink aggregation or single source buy-at-bulk. *Algorithmica*, 43(1–2):5–15, August 2005. Disponível em `http://www.springerlink.com/content/q3j33731u59x3211/fulltext.pdf`.