Eidgenössische
Technische Hochschule
Zürich

Ecole polytechnique fédérale de Zurich
Politecnico federale di Zurigo
Swiss Federal Institute of Technology Zurich

Chair of Computational Science

Prof. Dr. Peter Arbenz

Decoding Problem

Master Thesis

Introduction

This project consists in the study and the implementation of a parallel algorithm the task of which is to solve the so-called *decoding problem*. The initial problem is given by a large $n \times m$ matrix A over $\mathbb{F}_2$, with $n \gg m$, and a target vector $\mathbf{y}$. The solution of the problem is an m-vector $\mathbf{x}$ such that the Hamming weight of $A\mathbf{x} + \mathbf{y} \bmod 2$ is as small as possible. One can show that this problem reduces to the low Hamming weight codeword problem, which can be tackled with the help of Stern's algorithm and its variants (see, e.g., J. Stern [6], or, A. Canteaut [1]). This problem is at the heart of the McEliece public-key cryptosystem, as well as other cryptological problems of interest.

This minimization problem is related to the least-squares problem in the real or complex numbers. In large scale applications, i.e., for very large values of n, it can be (approximately) solved by the Lanczos algorithm [2]. Here, an implementation is required for the finite field $\mathbb{F}_2$. The purpose of this thesis is the implementations of Stern's algorithm for very large system sizes n.

The implementation will be based on the Trilinos framework [3].

This project will be executed in close collaboration with the Eidg. Dept. Verteidigung, Bevölkerungsschutz & Sport (VBS), Department of Information Security and Cryptology.

Tasks

- Get familiar with the required cryptology.
- Get familiar with the integer version of the Arnoldi or Lanczos algorithm, also known as Wiedemann algorithm.
- Get familiar with the parallel framework Trilinos [4, 7]
- Come up with a time-table for the master thesis.
- Implementation of the chosen algorithm.
- Validation and Benchmarking.
- Application to a *real world* example that will be determined later.

Requirements

- Good knowledge in C++.
- Some knowledge in cryptology.
- The attendance of a parallel computing course is very useful.
- Willing to work in an interdisciplinary environment.

Deliverables

The work is to be documented in a short and concise thesis. It must be written such that it is intelligible to a fellow-student.

The code should be written as clean as possible. It must be complemented by a short user's guide.

Presentation

At the end of the thesis the work is to be presented in a talk at a seminar of the Chair of Computational Science. The date of the talk will be determined later.

Contact

- Prof. Dr. Peter Arbenz, arbenz@inf.ethz.ch, Tel. 044 632 7432
- Dr. Gérard Maze, Institut für Mathematik, Universität Zürich. gmaze@math.uzh.ch, Tel. 044 635 5830.

References

[1] A. Canteaut. A new algorithm for finding minimum-weight words in large linear codes. In Colin Boyd, editor, *Cryptography and Coding*, volume 1025 of *Lecture Notes in Computer Science*, pages 205–212. Springer, Berlin, 1995.

[2] G. H. Golub and C. F. van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, MD, 3rd edition, 1996.

[3] M. A. Heroux, R. A. Bartlett, V. E. Howle, R. J. Hoekstra, J. J. Hu, T. G. Kolda, R. B. Lehoucq, K. R. Long, R. P. Pawlowski, E. T. Phipps, A. G. Salinger, H. K. Thornquist, R. S. Tuminaro, J. M. Willenbring, A. Williams, and K. S. Stanley. An overview of the Trilinos project. *ACM Transactions on Mathematical Software*, 31(3):397–423, 2005.

[4] M. A. Heroux and J. M. Willenbring. Trilinos Users Guide. Technical Report SAND2003-2952, Sandia National Laboratories, August 2003.

[5] L. Minder and A. Sinclair. The extended k-tree algorithm. In *Proceedings of the twentieth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA'09, pages 586–595, Philadelphia, PA, 2009. SIAM.

[6] J. Stern. A method for finding codewords of small weight. In Gérard Cohen and Jacques Wolfmann, editors, *Coding Theory and Applications*, volume 388 of *Lecture Notes in Computer Science*, pages 106–113. Springer, Berlin, 1989.

[7] The Trilinos Project Home Page. `http://trilinos.sandia.gov/`.

6. Januar 2011 C. Paglino/P. Arbenz