Bruno Buchberger et al. (Eds.)

Hagenberg Research

Springer

Hagenberg Research

Bruno Buchberger · Michael Affenzeller
Alois Ferscha · Michael Haller · Tudor Jebelean
Erich Peter Klement · Peter Paule
Gustav Pomberger · Wolfgang Schreiner
Robert Stubenrauch · Roland Wagner
Gerhard Weiß · Wolfgang Windsteiger
Editors

Hagenberg Research

Springer

Editors

Bruno Buchberger, Bruno.Buchberger@risc.jku.at
Michael Affenzeller, michael.affenzeller@fh-hagenberg.at
Alois Ferscha, ferscha@soft.uni-linz.ac.at
Michael Haller, haller@fh-hagenberg.at
Tudor Jebelean, Tudor.Jebelean@risc.jku.at
Erich Peter Klement, ep.klement@jku.at
Peter Paule, Peter.Paule@risc.jku.at
Gustav Pomberger, gustav.pomberger@jku.at
Wolfgang Schreiner, Wolfgang.Schreiner@risc.jku.at
Robert Stubenrauch, stubenrauch@softwarepark-hagenberg.com
Roland Wagner, rwagner@faw.jku.at
Gerhard Weiss, gerhard.weiss@scch.at
Wolfgang Windsteiger, Wolfgang.Windsteiger@risc.jku.at

A-4232 Hagenberg
Austria

ISBN 978-3-642-02126-8 e-ISBN 978-3-642-02127-5
DOI 10.1007/978-3-642-02127-5
Springer Dordrecht Heidelberg London New York

Library of Congress Control Number: 2009928635

ACM Computing Classification (1998): D.2, H.3, I.2, C.2, H.5, F.1

Cover design: KünkelLopka, Heidelberg

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Contents

Acknowledgement

This book was sponsored by

- Austrian Ministry of Science and Research (BMWF),
- Austrian Ministry for Transport, Innovation and Technology (BMVIT),
- Upper Austrian Government,
- Johannes Kepler University Linz (JKU),
- Community of Hagenberg,
- Raiffeisenbank Pregarten–Hagenberg.

In the preparation of this manuscript, the support in TEX-programming by Manuel Kauers was very much appreciated.

Hagenberg Research: Introduction

Bruno Buchberger

This book is a synopsis of basic and applied research done at the various research institutions of the Softwarepark Hagenberg in Austria. Starting with 15 coworkers in my Research Institute for Symbolic Computation (RISC), I initiated the Softwarepark Hagenberg in 1987 on request of the Upper Austrian Government with the objective of creating a scientific, technological, and economic impulse for the region and the international community. In the meantime, in a joint effort, the Softwarepark Hagenberg has grown to the current (2009) size of over 1000 R&D employees and 1300 students in six research institutions, 40 companies and 20 academic study programs on the bachelor, master's and PhD level.

The goal of the Softwarepark Hagenberg is innovation of economy in one of the most important current technologies: software. It is the message of this book that this can only be achieved and guaranteed long-term by "watering the root", namely emphasis on research, both basic and applied. In this book, we summarize what has been achieved in terms of research in the various research institutions in the Softwarepark Hagenberg and what research vision we have for the imminent future.

When I founded the Softwarepark Hagenberg, in addition to the "watering the root" principle, I had the vision that such a technology park can only prosper if we realize the "magic triangle", i.e. the close interaction of research, academic education, and business applications at one site, see Figure 1.

This principle proved to be quite successful: research pulls academic education and economic innovation, companies have a motivating and challenging influence on both research and the contents and implementation of curricula, and well trained graduates on all levels guarantee fresh energy for research and competitiveness of companies. In the meantime, this principle has been adopted widely to the extent that, recently (2008), EU President Barroso proclaimed the "Magic Triangle" as the building principle for the new "European Institute for Innovation and Technology" to be founded within the next few months. It is very fulfilling for me to see that this principle now receives such a prominent attention.

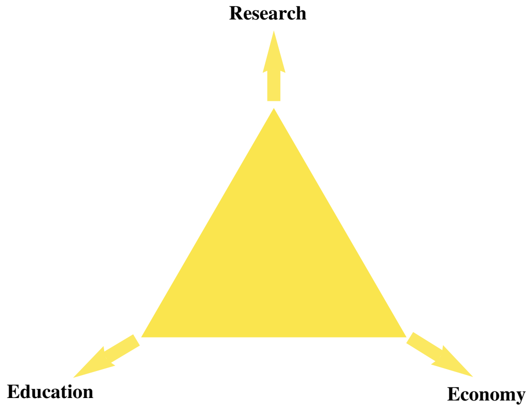

FIGURE 1 The Magic Triangle.

In this book, Hagenberg Research is summarized in various chapters that span the wide range of research topics pursued at the following research institutions in the Softwarepark Hagenberg:

- RISC (Research Institute for Symbolic Computation), the founding institute of the Softwarepark Hagenberg
- FAW (Institute for Application Oriented Knowledge Processing)
- FLLL (Department of Knowledge-Based Mathematical Systems, Fuzzy Logic Laboratorium Linz-Hagenberg)
- RIPE (Research Institute for Pervasive Computing)
- The Software Competence Center Hagenberg
- School of Informatics, Communication and Media, Upper Austria University of Applied Sciences, Research Center Hagenberg

The research strategy we pursue at the Softwarepark Hagenberg emphasizes the flow from formal logic, algorithmic mathematics, to software (and, to a lesser extent) hardware science. In my understanding, logic, mathematics, and software science form a coherent and indistinguishable magma of knowledge and methods (which I like to call the "thinking technology") and this is the strength from which we draw in the Softwarepark Hagenberg.

I am happy and fulfilled to see that this view is providing a solid basis for such a dynamic and future-oriented construct as the Softwarepark Hagenberg. This view also guided me as my personal strategy since the time of writing my PhD thesis in 1965, in which I introduced the theory of Gröbner bases (see [Buc65, Buc70]), which in the meantime became a powerful algorithmic tool for a constantly expanding range of applications in all situations where we have to deal with problems that can be cast in the language of non-linear polynomial systems. The coherent magma of logic, mathematics, and software science can be well demonstrated by the development of the field of Gröbner bases:

- The Gröbner bases method as an algorithmic method is based on a theorem (see [Buc65]) of pure algebra (the Theorem on the characterization of Gröbner bases by the zero-reducibility of the so called S-polynomials, see Section 2 on Gröbner Bases in Chapter I on symbolic computation).
- The proof of the main theorem of Gröbner bases theory, which was quite a challenge at the time of its invention, by recent advances in automated theorem proving in my Theorema Group can now be produced automatically (see [Buc04] and Chapter II on automated reasoning) to the extent that even the key *idea* of the theorem, S-polynomials, can be generated automatically.
- The Gröbner bases method can be applied in a growing number of seemingly quite distinct fields as, for example, coding theory and cryptography, robotics, systems and control theory, invariant theory, symbolics of combinatorial identities etc. (see again Section 2 in Chapter I). Interestingly, it also can be applied to automated theorem proving (notably geometrical theorem proving) and theorem invention and, by recent research in the Theorema Group (see Chapter II), to fundamental questions of software science like the automated generation of loop invariants of algorithms.

In this example, we see how the logic/mathematics/software science "magma" reaches out and bends back to itself in a constant movement of expansion and self-reference conquering higher and higher levels of understanding and methodology. This process, by what we know from Gödel's second theorem, does not have any limitation. Translating this to the "politics" of an institution like the Softwarepark Hagenberg: As long as we base our expansion and growth on research, there is no apparent limit to what we can achieve by our cooperative effort embedded into the international research community.

As the founder of the Research Institute for Symbolic Computation (Johannes Kepler University) and the founder and head of the Softwarepark Hagenberg (1987) I am proud to present the results of our joint research efforts in this book and I look forward to the next steps of our joint growth in intense interaction with the international research community. We will also be particularly happy to welcome our colleagues from all over the world at the research and conference facilities which we are currently expanding by generous grants from the Upper Austrian Government.

I also want to thank my colleagues in the Softwarepark Hagenberg research institutions for years of joint work and for their contributions to this book. My sincere thanks go to the Austrian and Upper Austrian Governmental Institutions and the various Austrian and EU research funding agencies and programs that made it possible to create the Softwarepark Hagenberg and to pursue our research.

Bruno Buchberger
Founder and Head of the Softwarepark Hagenberg

FIGURE 2 The Softwarepark Hagenberg.

References

[Buc65] B. Buchberger. *An Algorithm for Finding the Basis Elements in the Residue Class Ring Modulo a Zero Dimensional Polynomial Ideal*. PhD thesis, University Innsbruck, Mathematical Institute, 1965. German, English translation in: J. of Symbolic Computation, Special Issue on Logic, Mathematics, and Computer Science: Interactions. Volume 41, Number 3–4, Pages 475–511, 2006.

[Buc70] B. Buchberger. An Algorithmical Criterion for the Solvability of Algebraic Systems of Equations. *Aequationes mathematicae*, 4(3):374–383, 1970. German. English translation in: B. Buchberger, F. Winkler (eds.), Groebner Bases and Applications, London Mathematical Society Lecture Note Series, Vol. 251, Cambridge University Press, 1998, pp. 535–545.

[Buc04] B. Buchberger. Towards the Automated Synthesis of a Gröbner Bases Algorithm. *RACSAM (Rev. Acad. Cienc., Spanish Royal Academy of Science)*, 98(1):65–75, 2004.

Chapter I
Algorithms in Symbolic Computation

Peter Paule
Bruno Buchberger, Lena Kartashova, Manuel Kauers,
Carsten Schneider, Franz Winkler

The development of computer technology has brought forth a renaissance of algorithmic mathematics which gave rise to the creation of new disciplines like Computational Mathematics. Symbolic Computation, which constitutes one of its major branches, is the main research focus of the Research Institute for Symbolic Computation (RISC).

In Section 1, author P. Paule, one finds an introduction to the theme together with comments on history as well as on the use of the computer for mathematical discovery and proving. The remaining sections of the chapter present more detailed descriptions of hot research topics currently pursued at RISC.

In Section 2 the inventor of Gröbner Bases, B. Buchberger, describes basic notions and results, and underlines the principal relevance of Gröbner bases by surprising recent applications. Section 3, author F. Winkler, gives an introduction to algebraic curves; a summary of results in theory and applications (e.g., computer aided design) is given. Section 4, author M. Kauers, reports on computer generated progress in lattice paths theory finding applications in combinatorics and physics. Section 5, author C. Schneider, provides a description of an interdisciplinary research project with DESY (Deutsches Elektronen-Synchrotron, Berlin/Zeuthen). Section 6, author E. Kartashova, describes the development of Nonlinear Resonance Analysis, a new branch of mathematical physics.

1 The Renaissance of Algorithmic Mathematics

"The mathematics of Egypt, of Babylon, and of the ancient Orient was all of the algorithmic type. Dialectical mathematics—strictly logical, deductive mathematics—originated with the Greeks. But it did not displace the algorithmic. In Euclid, the role of dialectic is to justify a construction—i.e., an algorithm. It is only in modern times that we find mathematics with little or no algorithmic content. [...] Recent years seem to show a shift back to a constructive or algorithmic view point."

To support their impression the authors of [DH81] continue by citing P. Henrici: "We never could have put a man on the moon if we had insisted that the trajectories should be computed with dialectic rigor. [...] Dialectic mathematics generates insight. Algorithmic mathematics generates results."

Below we comment on various aspects of recent developments, including topics like numerical analysis versus symbolic computation, and pure versus applied mathematics. Then we present mathematical snapshots which—from symbolic computation point of view—shed light on two fundamental mathematical activities, *discovery* (computer-assisted guessing) and *proving* (using computer algebra algorithms).

1.1 A Bit of History

We will high-light only some facets of the *recent* history of algorithmic mathematics. However, we first need to clarify what algorithmic mathematics is about.

Algorithmic vs. Dialectic Mathematics

About thirty years ago P.J. Davis and R. Hersh in their marvelous book [DH81] included a short subsection with exactly the same title. We only make use of their example (finding $\sqrt{2}$) to distinguish between algorithmic and dialectic (i.e. non-algorithmic) mathematics. But to the interested reader we recommend the related entries of [DH81] for further reading.

Consider the problem to find a solution, denoted by $\sqrt{2}$, to the equation $x^2 = 2$.

Solution 1

Consider the sequence $(x_n)_{n\geq 1}$ defined for $n \geq 1$ recursively by

$$x_{n+1} = \frac{1}{2}\left(x_n + \frac{2}{x_n}\right),$$

with initial value $x_1 = 1$. Then $(x_n)_{n\geq 1}$ converges to $\sqrt{2}$ with quadratic rapidity. For example, $x_4 = \frac{577}{408} = 1.414215\ldots$ is already correct to 5 decimal places. Note, the algorithm can be carried out with just addition and division, and without complete theory of the real number system.

Solution 2

Consider the function $f(x) = x^2 - 2$ defined on the interval from 0 to 2. Observe that f is a continuous function with $f(0) = -2$ and $f(2) = 2$. Therefore, according to the intermediate value theorem, there exists a real number, let's call it $\sqrt{2}$, such that $f(\sqrt{2}) = 0$. Note, the details of the argument are based on properties of the real number system.

Solution 1 is algorithmic mathematics; solution 2 is the dialectic solution. Note that, in a certain sense, neither solution 1 nor solution 2 is a solution at all. Solution 1 gives us a better and better approximation, but no x_n gives us an exact solution. Solution 2 tells us that an exact solution exists between 0 and 2, but that is all it has to say.

Numerical Analysis vs. Symbolic Computation

Readers interested in the relatively young history of symbolic computation are referred to respective entries in the books [GCL92] and [vzGG99]. Concerning the first research journal in this field, [vzGG99] says, "The highly successful Journal of Symbolic Computation, created in 1985 by Bruno Buchberger, is the undisputed leader for research publication." So in 1981 when the book [DH81] appeared, symbolic computation was still at a very early stage of its development. This is reflected by statements like: "Certainly the algorithmic approach is called for when the problem at hand requires a numerical answer which is of importance for subsequent work either inside or outside mathematics."

Meanwhile this situation has changed quite a bit. Nowadays, symbolic computation and numerical analysis can be viewed as two sides of the same medal, i.e. of algorithmic mathematics. In other words, until today also symbolic computation has developed into a discipline which provides an extremely rich tool-box for problem solving outside or inside mathematics. Concerning the latter aspect, in view of recent applications, including some being described in the sections of this chapter, symbolic computation seems to evolve into a key technology in mathematics.