CENTER FOR
MACHINE PERCEPTION

CZECH TECHNICAL
UNIVERSITY IN PRAGUE

BACHELOR THESIS

Minimal Problem Solver Generator

Pavel Trutman

pavel.trutman@fel.cvut.cz

April 5, 2015

Available at
http://cmp.felk.cvut.cz/~trutmpav/theses/bsc-pavel-trutman.pdf

Thesis Advisor: Ing. Tomáš Pajdla, PhD.

Acknowledge grants here. Use centering if the text is too short.

Center for Machine Perception, Department of Cybernetics
Faculty of Electrical Engineering, Czech Technical University
Technická 2, 166 27 Prague 6, Czech Republic
fax +420 2 2435 7385, phone +420 2 2435 7637, www: http://cmp.felk.cvut.cz

Minimal Problem Solver Generator

Pavel Trutman

April 5, 2015

Text of acknowledgements. . .

Abstract

Text of abstract...

Resumé

Text of resumé...

Contents

Abbreviations

AHA! Some optional explanation before the list. Indentation can be set by the command `\setlength{\AbbrvIndent}{5em}`.

1D	one dimension(al)
2D, 3D, ...	two dimension(al), three dimension(al), two dimension(al), three dimension(al), two dimension(al), three dimension(al), two dimension(al), three dimension(al), ...
AAM	active appearance model
AI	artificial intelligence
ASM	active shape model
B-rep	boundary representation
BBN	Bayesian belief networks

1 Introduction

Here comes introduction.

2 Polynomial system solving

Firstly we review the state of the art algorithms for computing Gröbner basis. Better understanding of these algoritmhs helps us to more efficiently integrate them into polynomial solving algorithms based on Gröbner basis computation.

2.1 Buchberger's Algorithm

Buchberger's Algorithm was the first algorithm for computing Gröbener basis and it was invented by Bruno Buchberger.

The first and easy, but very inefficient, implementation of this algorithm says that we can extend a set F of polynomials to a Gröbner basis only by adding all nonzero remainders $\overline{S(f_i, f_j)}^F$ of all pairs in F into F. The pseudocode of this algorithm can be found as Theorem 2 in Section 2, §7 in [1]. Gröbner basis computed by this algorithm are often bigger than necessary.

2.2 F4 Algorithm

2.3 F5 Algorithm

3 Automatic generator

3.1 Reimplementation

3.2 Multiple eliminations solver

3.3 Removing unnecessary polynomials

3.4 Matrix partitioning

3.5 F4 strategy

4 Experiments

5 Conclusion

Bibliography

[1] David Cox, John Little, and Donald O'Shea. *Ideals, Varieties, and Algorithms : An Introduction to Computational Algebraic Geometry and Commutative Algebra.* Undergraduate Texts in Mathematics. Springer, New York, USA, 2nd edition, 1997. 4