Ordering in Solving Systems of Equations *

Kun Jiang Xiao-Shan Gao

Institute of Systems Science, Academia Sinica,
Beijing 100080, P.R. China
(kjiang, xgao)@mmrc.iss.ac.cn

Abstract

To use a "good" variable order is one of the effective ways to prevent the occurrence of large polynomials in an elimination algorithm. In this paper, we present an algorithm to find such a "good" order using algorithms from graph theory. Based on this algorithm, we present a new version of Wu-Ritt's zero decomposition algorithm, which allows different orders for different triangular sets in the decomposition formula.

Key Words: Variable ordering, Graph theory, Solving systems of equations.

1 Introduction

It is well known that Wu's method is very fast in proving geometry theorems[13, 14]. Many difficult theorems whose traditional proofs need an enormous a mount of human intelligence, such as Feuerbach's theorem, Morley's trisector theorem, etc., can be proved by computer programs based on Wu's method within seconds[2, 15]. This method firstly introduces a coordinate system so that the geometric entities and relations are turned into a system of equations, and then decides in an algorithmic manner whether the conclusion is valid on the solutions under some non-degenerate conditions. Therefore, Wu's method is essentially based on an algebraic elimination theory. One of the reason behind the success of Wu's method is that we may choose a nice order for the points in the geometry theorem. This order will lead to a variable order which will make the equation system easy to triangularize.

It is well-known that finding a "good" variable order is one of the effective ways to prevent the occurrence of large polynomials in a general elimination algorithm. So it is an effective strategy to find a nice order of variables before starting an elimination process. For instance, when using Wu's method to prove geometry theorems, one difficulty is the problem of reducibility. It has been noticed that deliberate choice of variable order can help avoiding or at least lessening reducibility difficulties [6].

There are two main approaches to find variable orders: one is at geometric level, the other is at algebraic level. Finding variable order in geometric level is actually a *geometric method* of equation

*This work is supported by a National Key Basic Research Project and by the CNSF under an outstanding youth grant(NO. 69725002)

solving, in which a complicated diagram is translated into a constructive form (using ruler and compass) which is easy to compute. This process is also called *geometric constraint solving.* There are mainly two geometric approaches to geometric constraint solving: the graph analysis approach [5, 10] and the rule-based approach [1, 3, 9]. In [6], a degree of freedom analysis method is proposed which is to disassemble the system of equations into spare parts, analyze the spare parts and the assemble them in a different way to find a nice order for geometric objects. Finding the variable order in algebraic level was not studied extensively. In [11], a heuristic method is proposed to find variable order mainly according to the degree of variables. This is a local method in that it does not provide global structure of the equation system.

In this paper, a graph approach is developed by analyzing the structural solvability of a system of equations. With this approach we can obtain some independent solvable parts in structural sense and the solving order among these parts. Consequently, the solving order of variables in the system of equations can be get. This algorithm is especially effective for large equation systems with hundreds of variables, like the equation systems from CAD systems [10].

As an application, we give a new version of Wu-Ritt's decomposition algorithm. Since Wu-Ritt's zero decomposition repeatedly does elimination for polynomial sets, we may use the graph algorithm to present a new version of Wu-Ritt's zero decomposition algorithm, which allows different order for different triangular sets in the decomposition formula.

2 DM-Composition of Bipartite Graph

We first recall the definition of the bipartite graph and related knowledge that play a central role in our approach. For an in-depth study, see [7, 8]. Let us assume that the system of equations to be solved is as follows:

$$\begin{cases} f_1(x_1, x_2, \ldots, x_n) = 0 \\ f_2(x_1, x_2, \ldots, x_n) = 0 \\ \qquad \vdots \\ f_n(x_1, x_2, \ldots, x_n) = 0, \end{cases} \tag{1}$$

where each f_i is a polynomial. In the above system of equations, it is possible that some variables disappear in some equations.

Let $G = (V, E)$ be a directed graph with vertex set V and edge set E. For two vertices u and v, we denote as " $u \longrightarrow v$ *on* G " (or simply, $u \longrightarrow v$), if and only if there exists a directed path from u to v on G.

Two vertices u and v belong to the same strongly connected component if and only if $u \longrightarrow v$ and $v \longrightarrow u$. The vertex set V is partitioned into strong connected components V_i, each of which determines the vertex induced subgraph $G_i = (V_i, E_i)$ of G, also called a strong component of G. Partial order $\prec$ can be defined on the family of strong components $G_i = (V_i, E_i)$ of G, or, in other words, on the family of subsets V_i of V, by

$$V_i \prec V_j \Longleftrightarrow v_j \longrightarrow v_i \text{ on } G \text{ for some } v_i \in V_i \text{ and } v_j \in V_j.$$

We also write $G_i \prec G_j$ if $V_i \prec V_j$.

A bipartite graph is a directed graph denoted by $B = (V^+, V^-; E)$ where the vertex set consists of two disjoint parts V^+ and V^-, and the edges are directed from V^+ to V^-. For $W^+(\subset V^+)$, we use $\Gamma(W^+)$ to denote the set of vertices in V^- adjacent to the vertices in W^+. A Matching M on B is a subset of E such that no two edges in M share a common vertex incident to them. A matching of maximum cardinality $|E|$ (the number of edges in M) is called a maximum matching. In case $|V^+| = |V^-|$, a matching that covers all the vertices is a complete matching. For M, $\partial^+ M$ stands for all initial vertices of all edges in M and $\partial^- M$ stands for all terminal vertices of all edges in M.

Based on the maximal matching, a unique decomposition of a bipartite graph into partially ordered irreducible bipartite subgraphs can be defined. This decomposition, due to Dulmage-Mendelssohn, will be referred to as the **DM-composition**, and the irreducible components $B_i = (V^+, V^-; E_i)$ as **DM-components**. An example of the **DM-decomposition** is illustrated in Figure 1 and Figure 2, where the bold edges are the maximum matching. An algorithm for the **DM-decomposition** is given below, where $G_M = (V^+ \cup V^-, \tilde{E}; S^+, S^-)$ is the auxiliary graph associated with a matching M and is defined as follows:

$$(v, w) \in \tilde{E} \Longleftrightarrow (v, w) \in E \ or \ (w, v) \in M,$$

and $S^+ = V^+ \backslash \partial^+ M$ and $S^- = V^- \backslash \partial^- M$ are called the *entrance* and *exit*.

Algorithm 1: The DM-decomposition of $B = (V^+, V^-; E)$

step 1 Find a maximum matching M on $B = (V^+, V^-; E)$ [7, 8].

step 2 Let $V_0 = \{v \in V^+ \cup V^- | w \longrightarrow v \ on \ G_M \ for \ some \ w \ in \ S^+\}$.

step 3 Let $V_\infty = \{v \in V^+ \cup V^- | v \longrightarrow w \ on \ G_M \ for \ some \ w \ in \ S^-\}$.

step 4 Let $V_i (i = 1, \dots, r)$ be the strongly connected components of the graph obtained from G_M by deleting the vertices of $V_0 \cup V_\infty$ and the edges incident thereto.

step 5 Define the partial order $\prec$ on $\{V_i | i = 0, 1, \dots, r, \infty\}$ as follows:

$$V_i \prec V_j \Longleftrightarrow v_j \longrightarrow v_i \ on \ G_M \ for \ some \ v_i \in V_i \ and \ v_j \in V_j.$$

In this algorithm, a maximum matching can be found in $O(|E|(min\{|V^+|, |V^-|\})^{1/2})$ [7, 8], where $|E|$ is the number of the edges and $|V^+|, |V^-|$ the number of V^+, V^- respectively. The execution time of **step 2** and **step 3** are direct proportional to $|E|$. Finding the strongly connected components can be done in $O((|V^+| + |V^-|)^2(|V^+| + |V^-| - 1))$. Therefore, the time complexity of the Algorithm 1 is $O((|V^+| + |V^-|)^3)$.

Note that the decomposition does not depend on the choice of maximum matching M. We call the component V_0(or B_0) the *minimal inconsistent part*, the component V_∞(or B_∞) the *maximum inconsistent part*, and the rest the *consistent part*. The **DM-decomposition** reveals the structure of a bipartite graph with reference to the maximum matching on it.

3 Generate Triangular Blocks

The basic idea of solving a system of equations is to triangularize the equation systems. Namely, the original system of equations is transformed into a triangular form that can be easily solved by sequentially solving a series of univariate polynomial equations. Note that the variable order is crucial for the

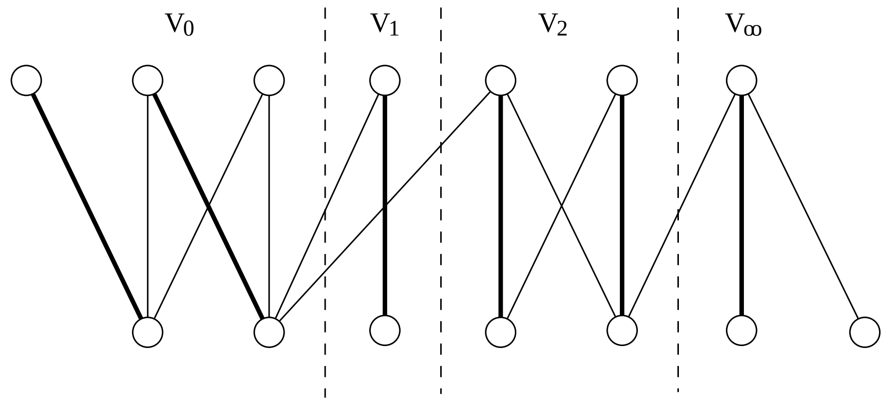

Figure 1: **DM-decomposition** of a bipartite graph

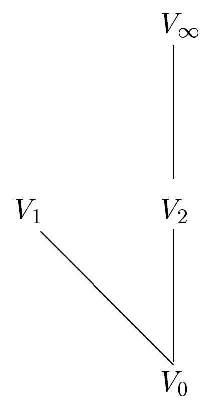

Figure 2: Partial order

triangularization. We can see that different orders will result in a great disparity in the complexity of solving. In general, a good order means that the original equations are already in a triangular block form, like the one shown in (2). In order to get a good order, we make use of the bipartite graph theory to analyze the structural solvability of the system of equations and decompose the system of equations into some subsets of equations, which are independent solvable parts named **triangular blocks**. With the **triangular blocks** and the partial order we can get a order of variables. This order is especially efficient in solving the system of nonlinear equations that consists of hundreds or thousands of variables and equations. In practice, this kind of system of equations often occur in CAD problem [10].

$$\left\{\begin{array}{l} f_1(x_{i_1},\ldots,x_{i_{t_1}})=0 \\ f_{t_1}(x_{i_1},\ldots,x_{i_{t_1}})=0 \\ f_{t_1+1}(x_{i_1},\ldots,x_{i_{t_1}},x_{i_{t_1}+1},\ldots,x_{i_{t_2}})=0 \\ \quad f_{t_2}(x_{i_1},\ldots,x_{i_{t_1}},x_{i_{t_1}+1},\ldots,x_{i_{t_2}})=0 \\ \qquad \vdots \\ f_{t_{b-1}+1}(x_{i_1},\ldots,x_{i_{t_1}},\ldots,x_{i_{t_{b-1}}+1},\ldots,x_{i_{t_b}})=0 \\ \qquad f_{t_b}(x_{i_1},\ldots,x_{i_{t_1}},\ldots,x_{i_{t_{b-1}}+1},\ldots,x_{i_{t_b}})=0. \end{array}\right. \tag{2}$$

3.1 Generating Triangular Blocks

We only consider a system of equations in the standard form (1), where x_i are variables and f_i are equations. Let F be the set of equations, X the set of all variables appearing in the equations and E the set of edges defined by the pairs(f_i, x_i) with $f_i \in F$ and $x_i \in X$ such that the variable x_i appears in the equation f_i. Then $B = (X, F; E)$ is the bipartite graph associated with the system of equations (see Example 1). This bipartite graph is called the bipartite graph representation of the system of equations.

Example 1 Consider the following system of equations:

$$\begin{cases} f_1(x_1, x_2, x_3) = 0 \\ f_2(x_2, x_3, x_5) = 0 \\ f_3(x_1, x_4) = 0 \\ f_4(x_3) = 0 \\ f_5(x_4, x_5) = 0. \end{cases}$$

The bipartite graph representation is shown in Figure 3.

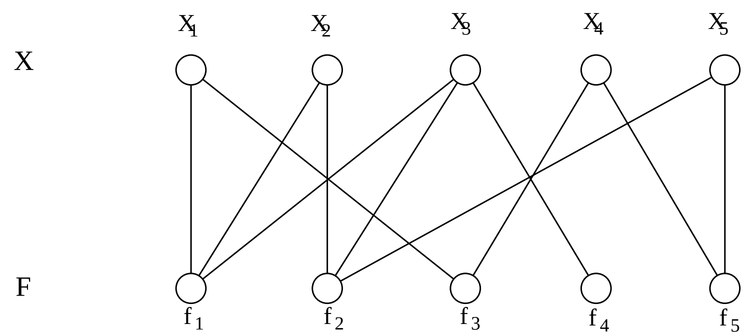

Figure 3: the Bipartite graph representation

Algorithm 2: Ordering in solving system of equations

Input: A system of equations.

Output: The solving order of variables of the system of equations.

step 1 Generate the bipartite graph representation $B = (X, F; E)$ for the system of equations.

step 2 Process the **DM-decomposition** of $B = (X, F; E)$ with **Algorithm 1**.

step 3 If $V_0 \neq \emptyset$ then the system of equations is under-determined and goto **step 5**.

step 4 If $V_\infty \neq \emptyset$ then the system of equations is over-determined. One or more equations in V_∞ should be deleted by the users to obtain a well-determined problem and goto **step 5**.

step 5 Output the variables in $\{V_i | i = 0, 1, \ldots, r, \infty\}$ according to the partial order defined on it.

According to the order of the variables, we can quickly solve the system of equations. In this algorithm, **step 1** can be done in $O(|E|)$ steps. The complexity of **step 5** is $O(|X|)$. Since $|E|$ is at most $O((|V^+| + |V^-|)^2)$, the complexity of Algorithm 2 is $O((|V^+| + |V^-|)^3)$,

Remark The **DM-decomposition** of the representation bipartite graph permits us to study the solvability of the system of equations. The system of equations is structurally solvable if and only if both V_0 and V_∞ are the empty sets. An **DM-component** $V_i, 1 \leq i \leq r$, in the consistent part corresponds to a **triangular block** that is structurally solvable and can not be decomposed further. It can be solved independently if the values of all the variables belonging to V_j such that $V_j \prec V_i$ are determined. The **triangular blocks** corresponding to the inconsistent parts, V_0 and V_∞, are not solvable. The problem corresponding to V_0 is under-determined (i.e. has more variables than equations) and that to V_∞ is over-determined (i.e. has fewer variables than equations). Note that the order of the variables from the same block can be given using the method presented in [11]. Note that the order of the variables from the same block can be given using the method presented in [11].

Example 2 Consider the following system of equations. The representation bipartite graph is shown in Figure 4. The **DM-decomposition** is shown in Figure 5 (The bold edges are maximum matching).

$$\begin{cases} f_1(x_1, x_3, x_7) = 0 \\ f_2(x_3, x_4, x_7) = 0 \\ f_3(x_2, x_5, x_6) = 0 \\ f_4(x_5, x_6) = 0 \\ f_5(x_2, x_5, x_7) = 0 \\ f_6(x_3, x_7, x_8, x_{12}) = 0 \\ f_7(x_4, x_8, x_9) = 0 \\ f_8(x_9, x_{10}, x_{11}) = 0 \\ f_9(x_{10}, x_{11}) = 0 \\ f_{10}(x_{11}, x_{12}) = 0 \\ f_{11}(x_6, x_{11}, x_{13}) = 0 \\ f_{12}(x_4) = 0 \\ f_{13}(x_9) = 0 \\ f_{14}(x_9) = 0. \end{cases}$$

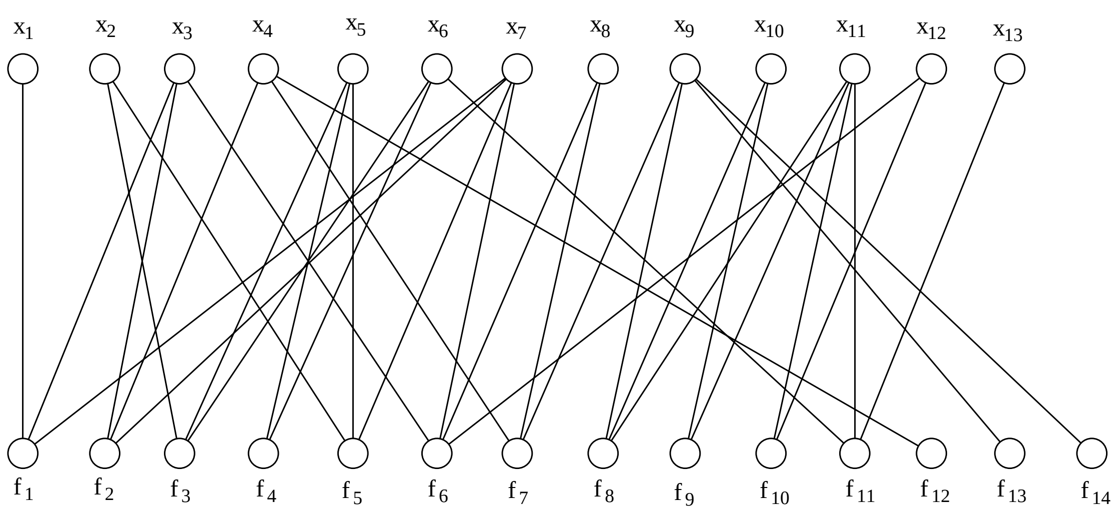

Figure 4: the Representation bipartite graph

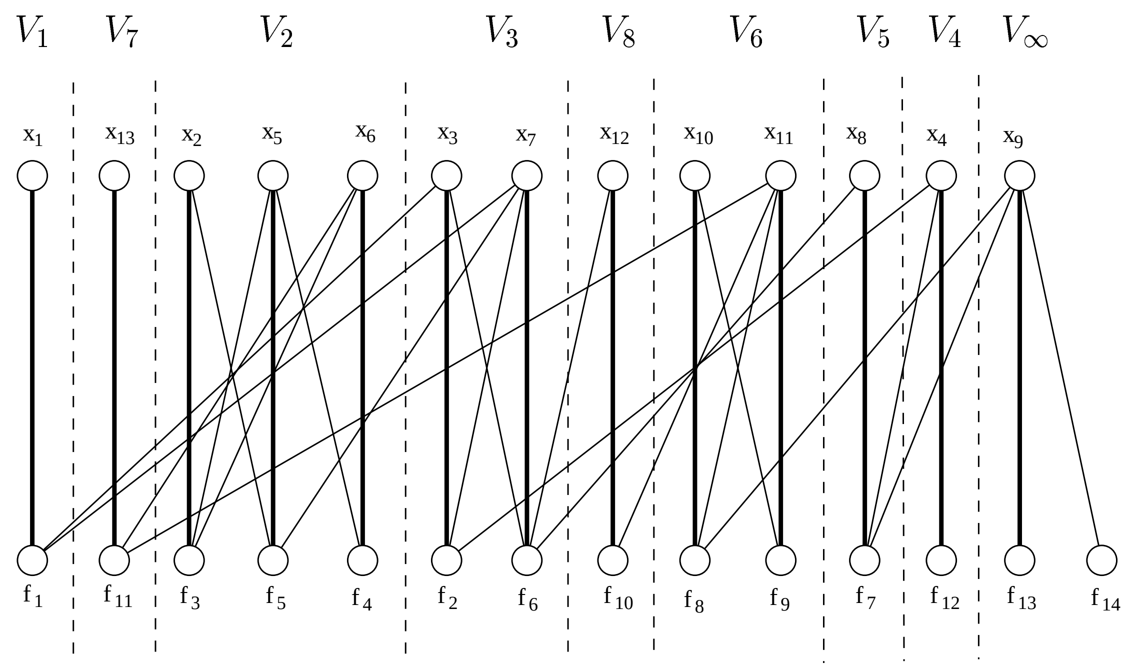

Figure 5: **DM-decomposition** of the representation bipartite graph

Since $V_\infty \neq \emptyset$ the system of equations is over-determined, i.e. the system of equations has more equations than variables. After getting rid of equation $f_{14}(x_9) = 0$, the solving order of variables is $\{x_9, x_4, x_8, x_{11}, x_{10}, x_{12}, x_7, x_3, x_6, x_5, x_2, x_{13}, x_1\}$.

3.2 Wu-Ritt's Zero Decomposition Algorithm with Mixed Order

Wu-Ritt's zero decomposition algorithm is used to decompose the zero set of a set of polynomial equations into the union of zero sets of polynomial equations in triangular form [13]. Precisely speaking, we have

Theorem(Wu-Ritt's Zero Decomposition Algorithm) For any given polynomial set PS, we may find a series of polynomial sets $TS_i, i = 1, \ldots, d$ in triangular form under one variable order such that

$$Zero(PS) = \bigcup_i Zero(TS_i/J_i)$$

where J_i is the product of initials of the polynomials in TS_i.

This method repeatedly uses a triangular procedure called well-ordering principle which takes a polynomial set PS as input and find a triangular polynomial set TS of PS [13].

Since the triangular procedure is sensitive to variable order, we propose that each time before using this triangular procedure we generate a new variable order with our algorithm. In this way, we have a different version of the decomposition algorithm.

Theorem (Wu-Ritt Zero Decomposition with Mixed Order) For a polynomial set PS, we may find a series of variable orders o_i such that

$$Zero(PS) = \bigcup_i Zero(TS_i/J_i)$$

where each TS_i is in triangular form under variable order o_i and J_i is the initial product of TS_i.

As an example, let us consider the $P3P$ problem [12, 4]: "Given the relative spatial locations of three control points, and given the angle to every pair of control points from an additional point called the Center of Perspective (C_P), find the lengths of the line segments joining C_P to each of the control points."

Let P be the Center of Perspective, and U, V, W the control points. Let $|PU| = x, |PV| = y, |PW| = z$ and $p = \cos\alpha, q = \cos\beta, r = \cos\gamma$. Introducing the following parameters: $A = \frac{b^2+c^2-a^2}{2}, B = \frac{c^2+a^2-b^2}{2}$, and $C = \frac{a^2+b^2-c^2}{2}$, we may find the following equation set PS

$$\begin{cases} x^2 + yzp - zxq - xyr - A = 0 \\ y^2 + zxq - yzp - xyr - B = 0 \\ z^2 + xyr - yzp - zxq - C = 0. \end{cases}$$

We need only to find solutions with correct physical meaning, i.e., we may assume $G \neq 0$, where

$$G = xyz\left(p^2-1\right)\left(q^2-1\right)\left(r^2-1\right)(Bx_1+C)(Ax_2+C)(Ax_3+B)$$

where x_1, x_2, x_3 are new introduced auxiliary variables. Applying Wu-Ritt's decomposition method [13] to (PS), we obtain a triangular set TS_1

$$\begin{cases} f_1 = I_1^2 x^8 + C_{16}x^6 + C_{14}x^4 + C_{12}x^2 + C_{10} \\ f_2 = xS_2 y - C_{20} \\ f_3 = xS_3 z - C_{30} \end{cases}$$

where $S_2 = (I_1I_2I_3)^4$, $S_3 = (I_1I_2I_4)^4$, $I_1 = p^2+q^2+r^2-2pqr-1$, $I_2 = Cq(rp-q) + Br(r-pq)$, $I_3 = Cp(rq-p) + Ar(r-pq)$, $I_4 = Bp(rq-p) + Aq(q-rp)$. Detailed coefficients of f_1, f_2, f_3 are omitted. Let $J = xI_1I_2I_3I_4$. From Wu-Ritt's Decomposition Theorem, we have

$$Zero(PS/G) \quad = \quad Zero(TS_1/J_1) \cup \bigcup_{i=1}^{4} Zero((PS, I_i)/I_1 \ldots I_{i-1}G)$$

where $TS_1 = \{f_1, f_2, f_3\}, J_1 = JG$. The first part of the above decomposition is the *main component* for the $P3P$ equation system and the last four zero sets correspond to the special or *degenerate cases*.

The three degenerate cases $I_i = 0, i = 2, 3, 4$ are similar. Let use consider $I_2 = 0$. Using the Gröbner basis program of Maple, we find that the polynomials set (PS, I_2) can be decomposed into two branches:

$$Zero((PS, I_2)/I_1G) = \bigcup_{i=1}^{2} Zero(GS_{2i}/I_1G) \text{ where}$$

$$GS_{21} = \begin{cases} I_2 \\ (pq-r)z^2 + rC \\ ry - qz \\ (pq-r)(x^2 - 2qxz - A) - Cqp \end{cases}$$

$$GS_{22} = \begin{cases} I_2 \\ rI_1x^2 + Cp(q-rp) + Ar(1-p^2) \\ rI_1(pq-r)(y^2 - 2rxy) - C_0 \\ (rp-q)z + (pq-r)y + (r^2+q^2-2pqr)x. \end{cases}$$

Here $C_0 = C\,(pr-q)\,(q^3-2pq^2r+qr^2-q+rp)+Ar\,(pq-r)\,(r^2+q^2-2pqr)$. Before using Wu-Ritt's algorithm to GS_{21} and GS_{22}, we may notice that GS_{21} and GS_{22} are already in triangular form under the variable orders: $z \prec y \prec x$ and $x \prec y \prec z$ respectively. This fact can be easily detected with our graph algorithm. So further decompositions for GS_{21} and GS_{22} should be carried out according to these two variable orders respectively.

4 Conclusion and Future Work

This paper proposes an approach to find variable order for solving a system of equations. We make use of the bipartite graph to decompose the system of equations into some **triangular blocks** in the structural sense. At the same time, we also obtain a partial order among these **triangular blocks**. According to this partial order, we sequentially output the variables in the **triangular blocks**, i.e. the variable order. Solving a system of equations with the order of variables obtained by our approach can speed up the solving procedure. Especially, in the case of system of equations that consists of hundreds or thousands of variables and equations, the proposed approach will be very efficient.

Our approach will lose effectiveness when all variables dependent on each other. In other words, these equations form one block. An interesting work is to break these blocks into smaller blocks. This has been done successfully for a large class of problems in geometric level [5, 10, 1, 3, 9]. In algebraic level, there is still no essential progress.

5 Acknowledgment

The authors want to thank Dr. Ding-Kang Wang for valuable discussions.

References

[1] B. Bruderlin, Using Geometric Rewriting Rules for Solving Geometric Problems Symbolically, *Theoretical Computer Science*, **116**, 291-303, 1993.

[2] S. C. Chou, *Mechanical Geometry Theorem Proving* ,D. Reidel Publishing Company, Dordrecht, Netherlands, 1988.

[3] X. S. Gao and S. C. Chou, Solving Geometric Constraint Systems I. A Global Propagation Approach, *Computer Aided Design*, **30**(1), 47-54, 1998.

[4] X. S. Gao and H. F. Cheng, On the Solution Classification of the "P3P" Problem, *Proceedings of the Third Asian Symposium on Computer Mathematics*, eds Z. B. Li, pp. 185-200, LanZhou University Press, 1998.

[5] C. Hoffmann, Geometric Constraint Solving in R^2 and R^3, in *Computing in Euclidean Geometry*, D.Z.Du and F.Huang(eds), Word Scientific, 1995, pp. 266-298.

[6] H. Li and M. Cheng, Automated Ordering for Automated Theorem Proving in Elementary Geometry-Degree of Freedom Analysis Method, *MM Research Preprints*, No. 18, pp.84-98, 1999.

[7] L. Lovász and M. D. Plummer, *Maching Theory*, Annals of Discrete Mathematics 29, North-Holland, Amsterdam, 1986.

[8] K. Murota, *Systems Analysis by Graphs and Matroids*, Algorithms and Combinatorics 3, Springer-Verlag, New York, 1987.

[9] G. A. Kramer, *Solving Geometric Constraints Systems: A Case Study in Kinematics*, MIT Press, 1992.

[10] J. Owen, Algebraic Solution for Geometry from Dimensional Constraints, in *ACM symp., Found of Solid Modeling*, ACM Press, Austin TX, pp. 397-407, 1991.

[11] D. Wang, An Implementation of the Characteristic Set Method in Maple. In: *Automated practical reasoning: Algebraic approaches*, Pfalzgraf, J., Wang, D. eds., Springer, Wien New York, pp. 187–201, 1995

[12] W. J. Wolfe, D. Mathis, C. Weber, and M. Magee, The Perspective View of Three Points, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **13**(1), 66–73, 1991.

[13] W. T. Wu, *Basic Principles of Mechanical Theorem Proving in Geometries*, Volume I: Part of Elementary Geometries, Science Press, Beijing (in Chinese), 1984.

[14] W. T. Wu, On the Decision Problem and the Mechanization of Theorem in Elementary Geometry, *Scientia Sinica*, **21**, , 159–172, 1987; Also in *Automated Theorem Proving: After 25 years*, A.M.S., Contemporary Mathematics, **29**, pp. 213-234, 1984.

[15] W. T. Wu, A Mechanization Method of Equations-solving and Theorem-proving, *Advances in Computing Research*, **6**, (ed. C. M. Hoffmann), pp. 103-138, 1992, JAI Press Inc, Greenwich, USA.