Department of Computer Science
University of Petrozavodsk

Automatic Generation of Test Problems and Experimental Analysis of Algorithms for Solving Non-Negative Linear Diophantine Equations

Kirill A. Kulakov, Dr. Dmitry G. Korzun

Abstract

One of the most significant requirements for creating software is a confirmation of the corrective implementation. Another significant requirement for that is experimental estimation of used recourses. In particular this estimation is useful for comparison with another algorithms.

In the research we study algorithms for solving non-negative linear diophantine equations (NLDE system). The target of this research is the technology for comprehensive testing, experimental and comparative analysis of these algorithms. The syntactic algorithm for solving associated with CF-grammars NLDE systems (ANLDE systems, introduced by D. G. Korzun) was used as a basic algorithm. The slopes algorithm for searching Hilbert basis (any NLDE system, introduced by M. Filgueiras, A.-P. Tomas) was used as an alternative for the syntax algorithm.

In this research we develop special algorithms for generation of test NLDE systems and their Hilbert bases. Our algorithms generate systems of some special classes. All algorithms based on received theorem about transformation any ANLDE system. Later we suppose to use the test systems for detailed experimental analysis of algorithm efficiency. Some preliminary experiments in this direction have been performed.

We implemented the software system that uses our test generation algorithms. Recently it performs testing of two NLDE-solving algorithms (syntax and slopes). The software has modular structure; work autonomously on a server for a long time. The software has independent from the available form of the algorithm (source code, executable program, etc.).

Algorithms for generating test ANLDE systems

System ANLDE is a special class of system NLDE which take the form (1):

$$\begin{pmatrix} 1 & \dots & 1 & 0 & \dots\dots\dots\dots\dots\dots\dots & & & & 0 \\ 0 & \dots & 0 & 1 & \dots & 1 & 0 & \dots\dots\dots\dots & 0 \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\ 0 & \dots\dots\dots\dots\dots\dots\dots & & & & 0 & 1 & \dots & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_m \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \dots\dots\dots\dots\dots\dots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_m \end{pmatrix} \tag{1}$$

All coefficients a_{ij} and components of solving of x_j are non-negative integers. Left-hand side matrix of coefficients of ANLDE system depends on a decomposition of quantity of unknowns onto subquantity's: $\{I_1, I_2, \dots, I_n\}$, $\bigcup\limits_{i=1}^{n} I_i = \{1, \dots, m\}$. For example system ANLDE (2).

$$\begin{cases} x_1 + x_2 & = & 2x_3 + x_4 \\ x_3 + x_4 & = & 3x_1 \end{cases} \tag{2}$$

Hilbert basis is a multitude of minimal indecomposable solutions of system. Hilbert basis for example (2) is (3).

$$\mathcal{H} = \left\{ \begin{pmatrix} 1 \\ 5 \\ 3 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 4 \\ 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 3 \\ 1 \\ 2 \end{pmatrix}, \begin{pmatrix} 1 \\ 2 \\ 0 \\ 3 \end{pmatrix} \right\} \tag{3}$$

For testing experimental and comparative analysis we must generate system NLDE and it's Hilbert basis. Let $S^{(0)}(I, A)$ — be a random system ANLDE $\mathrm{E}(I)x = Ax$. Lets examine a transformation of system ANLDE $S^{(0)}$ which will execute step-by-step

$$S^{(0)} \to S^{(1)} \to \dots \to S^{(r)}$$

After each step of transformation immediate system will have one smaller equation.

Let $S^{(l)}$ — be a current system $\widetilde{A}^{(l)}x = \mathbb{O}$. Sum up all equations:

$$\sum_{j=1}^{m} c_j x_j = 0, \qquad \text{where } c_j = \sum_i \widetilde{A}^{(l)}_{ij}. \tag{4}$$

Transformation $S^{(l)} \to S^{(l+1)}$ is executed if at least one coefficient $c_j = -1$. In this case we find such equation i_0

$$K = \left\{ k \mid \widetilde{A}^{(l)}_{i_0 k} = -1 \text{ and } \widetilde{A}^{(l)}_{ik} = 0 \text{ for all } i \neq i_0 \right\} \neq \varnothing.$$

Using equation i_0 we express unknowns x_k, $k \in K$:

$$\sum_{k \in K} x_k = \sum_{k \notin K} \widetilde{A}^{(l)}_{i_0 k} x_k = T_{l+1}(x).$$

The system $S^{(l+1)}$ is constructed as multitude of all equations from system $S^{(l)}$ except equation i_0.

If the last system $S^{(r)}$ consists of one equation, then the system is given by equation:

$$\sum_{j\in I_l} x_j = \sum_{j=1}^{m} a_{lj}x_j \tag{5}$$

where l — residuary equation of initial system. If $p = n - r > 1$, then all coefficients $c_j \geqslant 0$. So if $c_j > 0$ then $x_j = 0$. For the coefficients $c_j = 0$ system is given by:

$$\sum_{i\in I_k} x_i = \sum_{j\in J_k} x_j,\ k = \overline{1,p}, \tag{6}$$

where $\bigcup\limits_{k=\overline{1,p}} I_k = \bigcup\limits_{k=\overline{1,p}} J_k$. Decomposition I, J — decomposition for system $S^{(r)}$ where $x_j = 0$ if $c_j > 0$.

Using this transformation we prove the next theorem:

Theorem 1 *Consider any ANLDE system of n equations with m unknowns:*

$$\mathrm{E}(I_1, \ldots, I_n)x = Ax. \tag{7}$$

The problem of searching Hilbert basis of system (7) is reduced to problem of searching Hilbert basis either of equation (5) or of system (6).

We developed and implemented 3 algorithms for generation of tests systems of special classes of system ANLDE. We also introduced the algorithm for generating tests systems of full class of system ANLDE (see figure 1). Each algorithm has been proved based on Theorem 1.

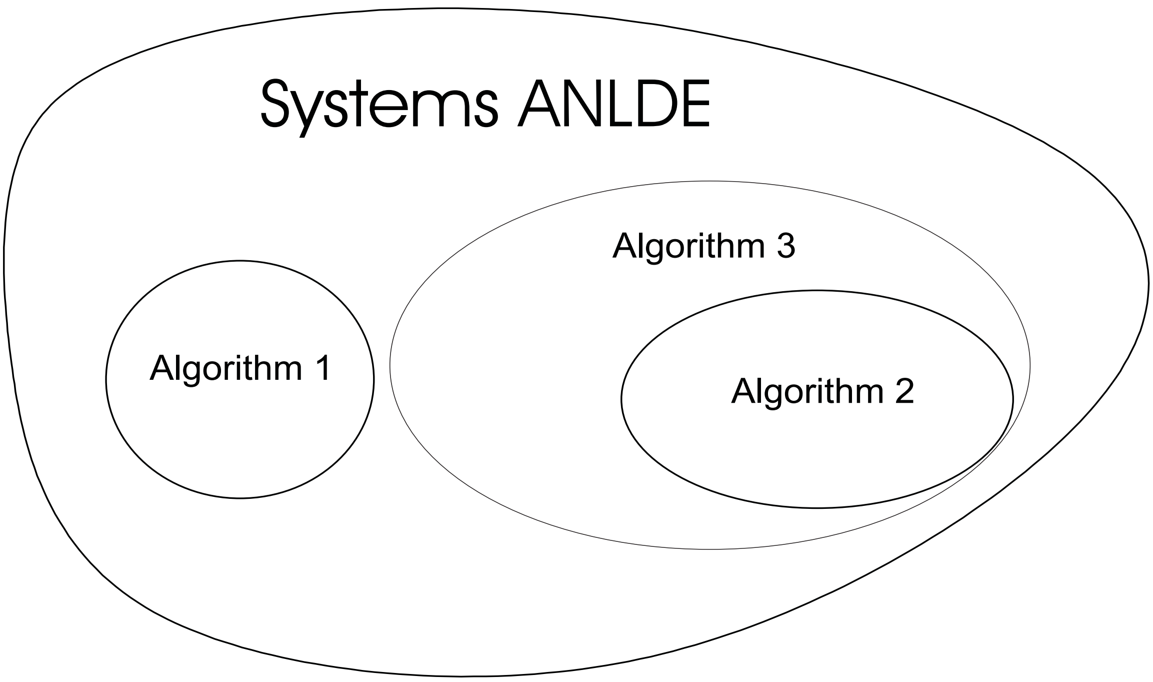

Figure 1: Decomposition systems ANLDE at classes

Software for testing and experimental analysis

The software for testing and collection of statistics on solving algorithm working has the modular structure (see figure 2). The main module controls the execution of all other modules. When the solving module starts working the efficiency analysis module collects statistics. After the end of solving module working the solving check module compares Hilbert bases and gives a result: correct solving or incorrect.

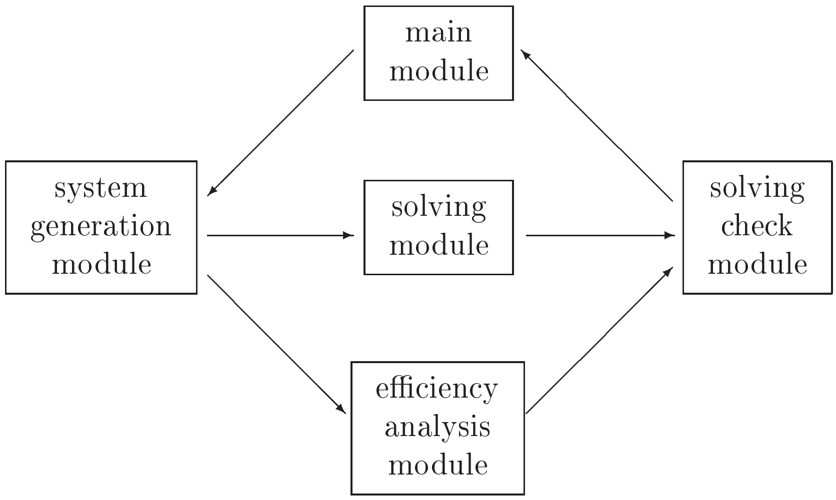

Figure 2: Program work scheme

Testing and experimental analysis

In experimental part of work we test implementation of syntax algorithm for solving systems ANLDE (anlde). We generated over 1.5 million test systems and anlde solved all systems.

In the next part of experimental analysis we compared anlde and implementation of algorithm for solving system NLDE introduced by M. Filgueiras, A.-P. Tomas (slopessys). For experimental analysis we generated 9500 systems and offered for solving. Due to slopessys has a limitation on system dimensions, we created systems with smaller dimensions but slopessys couldn't solving all systems.

As a result of experimental analysis we created a plot of dependence of solving time on system dimensions, for example dependence of solving time on a number of vectors in basis (see figure 3, 4).

Conclusion

In this research we received the following results:

— introduced and proved the theorem of transformation of ANLDE system;

— developed and implemented 3 algorithms for generation of test systems of special classes of system ANLDE;

— introduced the algorithm for generating test systems of full class of system ANLDE;

— implemented software system for testing and collection of statistics on solving algorithms working;

— performed preliminary experimental analysis of algorithm efficiency.

Dependence of solving time on a number of vectors in basis

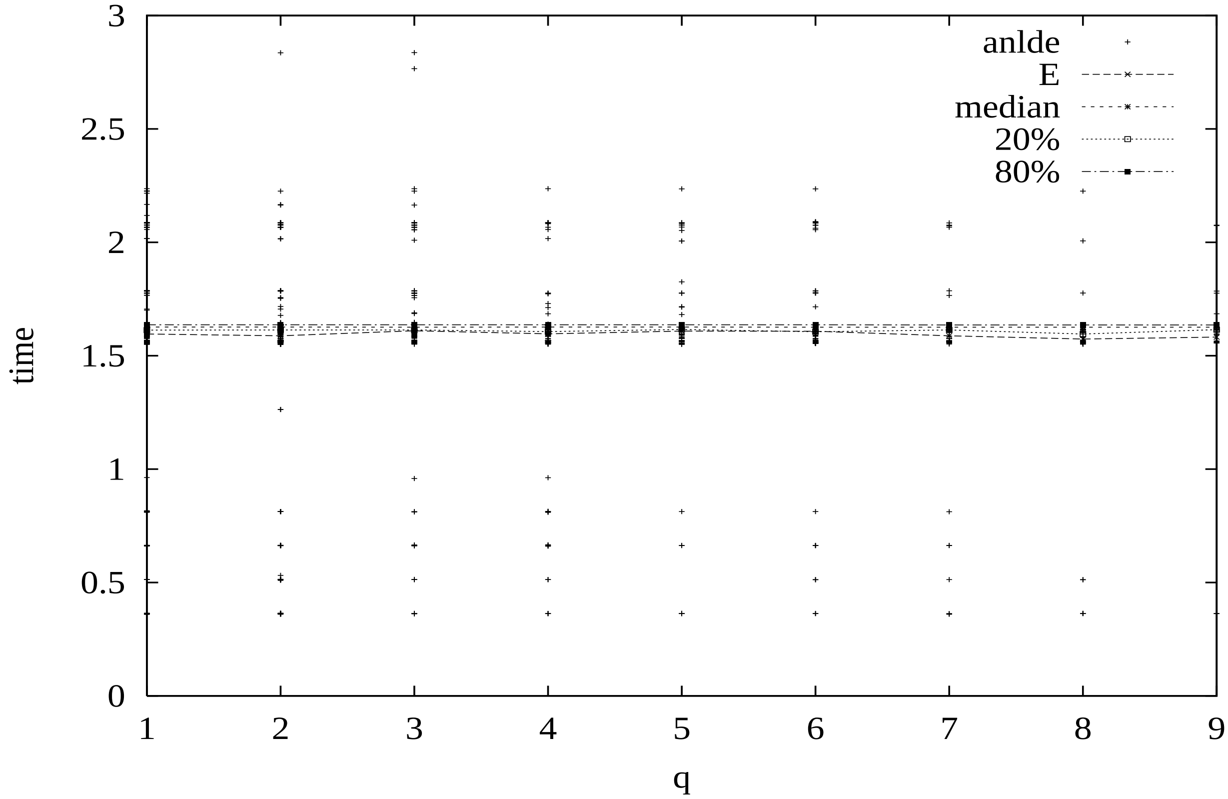

Figure 3: anlde solving program

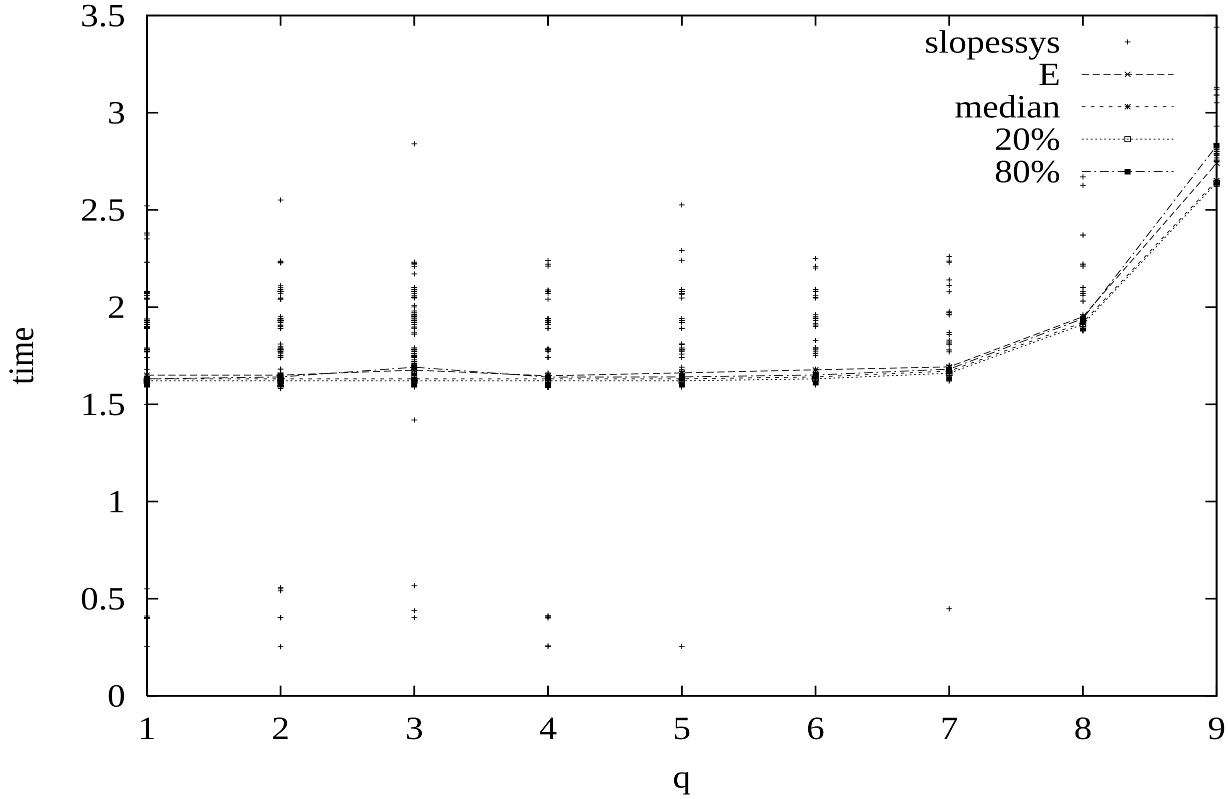

Figure 4: slopessys solving program