Modified general recursion algebraic method of the linear control systems reachable sets computation

Andrey F. Shorikov[1]
afshorikov@mail.ru

Alexandr Y. Goranov[2]
goranovayu@mail.ru

1–Ural Federal University (Yekaterinburg, Russia)
2–JSC "Scientific and Production Association of Automatics" (Yekaterinburg, Russia)

Abstract

In this paper, problem of reachable set computation of the linear discrete-time controlled system is considered. It is supposed that controlled plant dynamics is described by the vector recurrence equation. It is assumed that the plant initial condition and its control parameters are constrained by the sets, which are convex, closed and limited polyhedrons with final number of vertices. The description of the modified method of reachable set computation is provided and the comparative analysis of the received results with an initial general recurrent algebraic method is made.

Keywords: Convex Hull, Linear Programming, Divide-and-Conquer, Reachable Set, Optimal Control.

1 Introduction

Nowadays, a lot of attention is paid to the solution of optimal control problems of discrete-time dynamical systems with terminal functional of quality. From [Chernousko94, Krasovskii68, Tyulyukin93, Shorikov97] it is known that for solving problem of this kind it is rational to compute the reachable sets of all terminal (final) states. The availability of reachable set allows to significantly simplify the solution of a optimal open-loop control problem.

Reachable set computation methods can be divided into two main groups. The first group is creation of the exact reachable sets containing all admissible states of dynamical system to which it can be steered [Lasserre91, Tyulyukin93, Shorikov97]. The second group is approximating methods assuming considerable reduction of calculation time, however giving only an approximate estimation of reachable set [Girard05, Kurzhanski02, Chernousko94].

This paper offers the modified method of exact reachable set computation for discrete-time dynamical system, which is based on the general recurrent algebraic method described in work [Shorikov97].

2 Properties of controlled linear systems reachable sets

Let us consider the given integer-valued period of time $t \in \overline{0,T} = \{0, 1, \ldots, T\}$, $T > 0$, $T \in \mathbb{N}$ ($\mathbb{N}$ is the set of all natural numbers) the class of the linear controlled systems with dynamics that is described by the discrete-time vector-matrix recurrence equation:

In: G.A. Timofeeva, A.V. Martynenko (eds.): Proceedings of 3rd Russian Conference "Mathematical Modeling and Information Technologies" (MMIT 2016), Yekaterinburg, Russia, 16-Nov-2016, published at http://ceur-ws.org

$$x(t+1) = A(t)x(t) + B(t)u(t), \tag{1}$$

where $x(\cdot) \in \mathbb{R}^n$ is the state vector (a phase vector); $u(\cdot) \in \mathbb{R}^p$ is the control (input) vector; $A(\cdot) \in \mathbb{R}^{n\times n}$ is the state matrix; $B(\cdot) \in \mathbb{R}^{n\times p}$ is the input matrix.

Let us consider that the vector of the initial state and the vector of admissible controls satisfy the set geometric constraints

$$x(0) \in \mathbf{X}(0) \subset \mathbb{R}^n; \tag{2}$$

$$u(t) \in \mathbf{P}(t) \subset \mathbb{R}^p, \tag{3}$$

where sets $\mathbf{X}(0)$ and $\mathbf{P}(t)$ are convex, closed and limited polyhedrons (i.e. convex polytopes) with final number of vertices in spaces $\mathbb{R}^n$ and $\mathbb{R}^p$, respectively.

Thus, the problem is stated as the set constraints (2), (3) to define a set of all possible phase states $\mathbf{G}(0, \mathbf{X}(0), T)$ to which the controlled system (1) can be transferred [Krasovskii68, Tyulyukin93] at time T

$$\begin{aligned} \mathbf{G}(0, \mathbf{X}(0), T) = \{x(T) : \ & x(T) \in \mathbb{R}^n, \\ & x(t+1) = A(t)x(t) + B(t)u(t),\ t \in \overline{0, T-1}, \\ & x(0) \in \mathbf{X}(0),\ u(t) \in \mathbf{P}(t)\}. \end{aligned} \tag{4}$$

Pair $(\mathbf{X}(0), 0)$ is understood as the a set of all admissible initial states of system at time $t = 0$. The reachable set (4) has the following main properties [Chernousko94, Shorikov97]:

1. Reachable set (4) is a convex polytope with the finite number of vertices for each time point $t \in \overline{0, T}$;

2. Evolutionary (semi-group) property of reachable sets:

   $$\mathbf{G}(0, \mathbf{X}(0), T) = \mathbf{G}(t, \mathbf{G}(0, \mathbf{X}(0), t); T),$$

   where $\mathbf{G}(0, \mathbf{X}(0), T)$ – the reachable set at time point corresponding to pair $(\mathbf{X}(0), 0)$.

3 Generalized recurrent method of the reachable sets computation

In the works [Tyulyukin93, Shorikov97] it is shown that the reachable set represents a convex polytope with the finite number of vertices in $\mathbb{R}^n$. In this method reachable set computation leads to solving of the sequence of single-step reachable set computation

$$\mathbf{G}\left(0, \mathbf{X}(0); T\right) = \mathbf{G}(t, \mathbf{G}_+(t); T),\ t \in \overline{1, T-1}, \tag{5}$$

where $\mathbf{G}_+(t) = \mathbf{G}\left(0, \mathbf{X}(0); t\right)$ is the reachable set at time t, corresponding to the pair $(\mathbf{X}(0), 0)$, which is a convex polytope with the finite number of vertices in $\mathbb{R}^n$.

In this case if we can realize reachable set computation only on one step forward, it is possible to obtain the final reachable set (in terminal time point), which is only determined by reachable set on previous step:

$$\mathbf{G}(0, \mathbf{X}(0); T) = \mathbf{G}(T-1, \mathbf{G}_+(T-1); T).$$

It is known that the representation of convex polytope can be carried out as the description of all his vertices on the one hand and as the description of basic hyperplanes (linear inequalities) on the other [Chernikov68, Shorikov97, Ziegler98].

Hence, for the solving of basic auxiliary problem we rely on the generalized recurrent algorithm [Shorikov97].

The set of the feasible states $x(t+1)$ of the considered controlled system (1), comprising all vertices of reachable set $\mathbf{G}(t, \mathbf{G}_+(t); t+1)$ at time $(t+1)$, $t \in \overline{0, T-1}$, corresponding to the pair $\mathbf{G}_+(t) \in \overline{0, T-1} \times 2^{\mathbb{R}^n}$, is defined according to the following algorithm.

Step 1. Forming of the set $\mathbf{\Gamma}_n(\mathbf{G}_+(t))$ of all vertices of polytope $\mathbf{G}_+(t)$.

Step 2. Forming of the set $\mathbf{\Gamma}_p(\mathbf{P}(t))$ of all vertices of polytope $\mathbf{P}(t)$.

Step 3. Defining the following sets characterizing free and controlled motion of the system (1) – (3).

$$\widehat{\mathbf{G}}_x(t+1) = \{\widehat{x}(t+1): \ \widehat{x}(t+1) = A(t)x(t), \ x(t) \in \mathbf{\Gamma}_n(\mathbf{G}_+(t))\},$$
$$\widehat{\mathbf{G}}_u(t+1) = \{\widehat{y}(t+1): \ \widehat{y}(t+1) = B(t)u(t), \ u(t) \in \mathbf{\Gamma}_p(\mathbf{P}(t))\},$$
$$\widehat{\mathbf{G}}_+(t+1) = \{\widehat{v}(t+1): \ \widehat{v}(t+1) = \widehat{x}(t+1) + \widehat{y}(t+1),$$
$$\widehat{x}(t+1) \in \widehat{\mathbf{G}}_x(t+1), \ \widehat{y}(t+1) \in \widehat{\mathbf{G}}_u(t+1)\},$$

where $\widehat{\mathbf{G}}_+(t+1)$ is the Minkowski sum of the sets $\widehat{\mathbf{G}}_x(t+1)$ and $\widehat{\mathbf{G}}_u(t+1)$ [Ziegler98], which are the set of points, among which are both internal and extreme points.

Step 4. For the defined set

$$\widehat{\mathbf{G}}_+(t+1) = \{\widehat{v}^{(i)}(t+1)\}_{i\in\overline{1,m}} \subset \mathbf{G}(t, \mathbf{G}_+(t), t+1)$$

we define the set of all its vertices

$$\mathbf{\Gamma}_n(\mathbf{G}_+(t+1)) = \{\widehat{v}^{(i)}(t+1)\}_{i\in\overline{1,k}}, \ (k \leq m).$$

In [Tyulyukin93] it is shown that the following statement holds:

$$\mathrm{conv}\Big(\widehat{\mathbf{G}}_+(t+1)\Big) = \mathbf{G}(t, \mathbf{G}_+(t); t+1) = \mathbf{G}_+(t+1).$$

Thus, the vertices of the set $\mathbf{G}_+(t+1)$ define the reachable set of controlled system (1) − (3) at time $(t+1)$. That reachable set is the convex polytope in $\mathbb{R}^n$.

4 Reachable set vertices search problem

In the works [Tyulyukin93, Shorikov97] it is shown that the set vertices search problem is reduced to the solution of m linear mathematical programming problems. In accordance with this fact, we are formulated the following statement of the optimization problem.

For fixed $i = \overline{1,m}$ and a set of variables $\lambda_j, \ \ j = \overline{1,m}, \ \ j \neq i, \ \ \lambda_j \in \mathbb{R}$, it is required to solve the following linear mathematical programming problem

$$\begin{gathered} f = \sum_j \lambda_j \to \min\,(\max\,) \\ \sum_j \lambda_j \widehat{v}^{(j)}(t+1) = \widehat{v}^{(i)}(t+1), \\ \sum_j \lambda_j = 1, \ \lambda_j \geqslant 0. \end{gathered} \tag{6}$$

For the solving of linear mathematical programming problem we will use a simplex-method with inverse matrix [Yudin69]. It is known [Papadimitriou85, Chernikov68] that for check of consistency of system (6) it is enough to consider only the first stage of a simplex-method, that is to search the reference basis admissible decision if it exists. For finding of the reference basis admissible decision we will use artificial basis technique. It follows that the choice of cost function f in the problem (6) is of no importance as its choice does not affect at the search of the reference basis admissible decision in any way.

At the first stage of a simplex-method the simplex table is a matrix, the coefficients of which on the last row can be considered as assessment of substitution of the appropriate columns [Yudin69], i.e. has the form:

$$M = \begin{pmatrix} a_{11} & \dots & b_1 & \dots & a_{1m} \\ a_{21} & \dots & b_2 & \dots & a_{2m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ a_{n1} & \dots & b_n & \dots & a_{nm} \\ 1 & \dots & 1 & \dots & 1 \\ \sum\limits_{k=1}^{n} a_{k1} + 1 & \dots & \sum\limits_{k=1}^{n} b_k + 1 & \dots & \sum\limits_{k=1}^{n} a_{km} + 1 \end{pmatrix},$$

The column b_k, $k = \overline{1,n}$ in the matrix M corresponds to coordinates of the fixed checked point $\widehat{v}^{(i)}(t+1)$, remaining columns $\widehat{v}^{(j)}(t+1)$ correspond to coordinates of points a_{kj}, $k = \overline{1,n}$, $j = \overline{1,m}$, $j \neq i$.

Further, there are two solutions of the linear mathematical programming problem:

1. The basis admissible solution is not found. It follows that the checked point corresponding to a vector $\widehat{v}^{(j)}(t+1)$ is the extreme point of set $\widehat{\mathbf{G}}_+(t+1)$ and, therefore, is the vertex of reachable set $\mathbf{G}(t, \mathbf{G}_+(t); t+1)$.

2. The basis admissible solution is found. In this case the checked point can be presented in the form of a convex combination of basis vectors B_i, $i \leqslant n+1$, therefore, this point is not the vertex of reachable set $\mathbf{G}(t, \mathbf{G}_+(t); t+1)$.

Thus, when the solutions of m linear mathematical programming problems are found, we computed set of all vertices $\mathbf{\Gamma}_n(\mathbf{G}_+(t+1))$ which will also be the reachable set $\mathbf{G}(t, \mathbf{G}_+(t); t+1)$ of dynamical system (1) – (3).

5 Modified method of the reachable sets computation

Now we present a modified version of the initial general recurrent method that is often much faster. This modified algorithm of reachable set computation relies on the ideas of "divide-and-conquer" algorithm, described in [Pardalos95, Preparata85]. This algorithm is often the main alternative to iterative algorithms, which example is the initial recurrent method.

"Divide-and-Conquer" is possible to present by means of the following consecutive steps:

1. Division of a problem into sub-problem, usually of a smaller size;

2. The solution of each of sub-problems (directly, if they are of rather small volume; differently if recursively breaking into smaller parts);

3. Combination of the received solutions of sub-problems.

For realization of this method in this paper the following step-by-step algorithm is offered. Using this algorithm it is possible to define all extreme points of reachable set:

The working set of points $\mathbf{X} = \{x^{(i)}\}_{i \in \overline{1,m}}$ is formed from the considered initial set of reachable set points $\widehat{\mathbf{G}}_+(t+1) = \{\widehat{v}^{(i)}(t+1)\}_{i \in \overline{1,m}} \subset \mathbb{R}^n$.

Then, it is necessary to compute the smallest multidimensional parallelepiped with the side parallel to the coordinate axes, which contains this set. Points of the set are sorted as increase in their distance from the multidimensional parallelepiped center, based on following values

$$\delta^{(i)} = \Big\| \max\Big\{ \frac{x_k^{(i)} - d_k}{r_k} \Big\} \Big\|, \; k \in \overline{1,n}, \; i \in \overline{1,m}.$$

where d_k is a coordinates of the center R; r is a vector, which components are lengths of the parallelepiped R sides; $\|\cdot\|$ is a designation of Euclidean norm.

On the next stage of the algorithm we will assume that first $k+1$ elements of sorted set $\mathbf{X}_s$ are extreme points of reachable set since these points have longest distance from center of parallelepiped R. Taking it into consideration we move these points to the set of pretender-points $\mathbf{X}_{pr}$.

Then, for each point $i = k+2, \ldots, n$ of sorted set it is necessary to find the solution of linear mathematical program (6). Thus, if the point $x_s^{(i)}$ is the vertex of polytope $\mathbf{X}_{pr} \bigcup x_s^{(i)}$, it is moved to set of pretender-points $\mathbf{X}_{pr}$, otherwise, it is excluded and does not participate in further work of an algorithm (in initial time point contains the two first points of the sorted set). Therefore, we solve the linear program of significantly smaller size than the algorithm presented in [Tyulyukin93, Shorikov97].

Since $\mathbf{X}_s$ becomes empty set, it is necessary to solve linear mathematical programming (6) for each pretender-point $x_s^{(i)} \in \mathbf{X}_{pr}$.

Algorithm action comes to the end after check of all points of the set $\mathbf{X}_{pr}$.

6 Comparative analysis of the original method and its modification

For the description of the modified recurrent algorithm the reachable set computation of linear discrete-time dynamical systems modeling has been carried out in the software environment MATLAB 8.4 (R2014a). The linear discrete-time model has served as an example

$$x(t+1) = Ax(t) + Bu(t),$$

where state matrix and control matrix are

$$A = \begin{pmatrix} 0 & 1 & 0 \\ -2 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}, \; B = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix},$$

and the vector of an initial states set and the vector of admissible controls are presented in the following form:

$$x = (x_1,\ x_2,\ x_3) \in \mathbb{R}^3;$$
$$x_1(0) = -0,2;\ x_2(0) = 0,2;\ x_3(0) = 0;$$
$$u = (u_1,\ u_2) \in \mathbb{R}^2;$$
$$u_1 \in [1, -1];\ u_2 \in [1, -1].$$

Modeling results of the reachable set computation for this system are presented in the Figure 1.

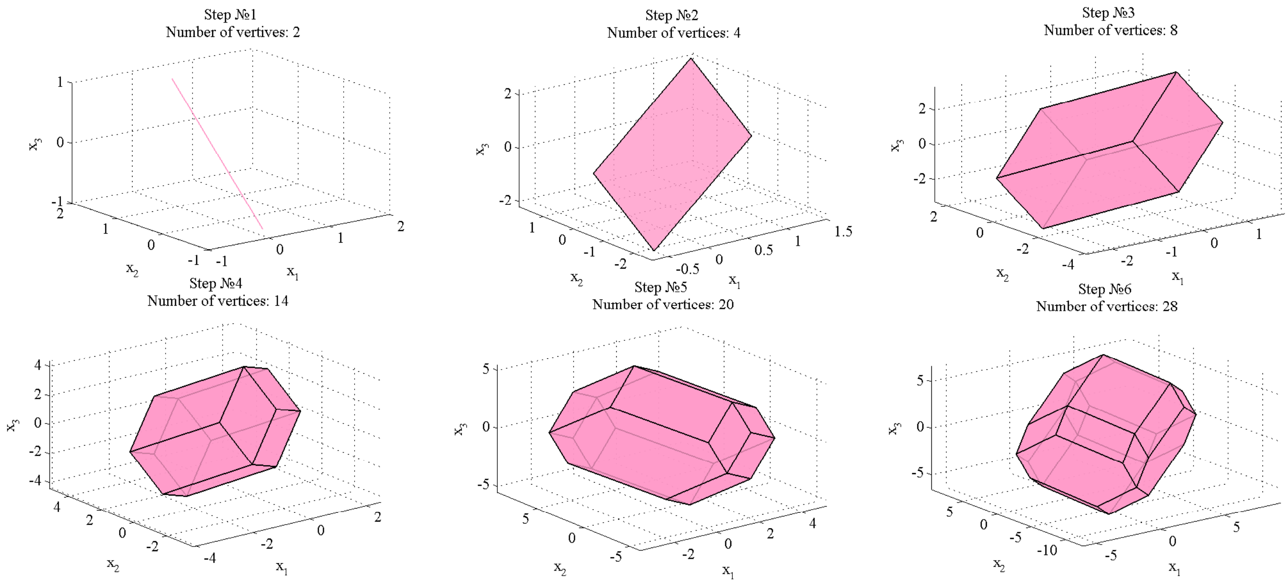

Figure 1: Reachable sets

For the more complete quality assessment of convex hull computation problem we solved that problem for two typical sets of random points:

1. Type Normal is the random set of points with the normal law distribution (expected mean $M_1 = 0$, root-mean-square deviation $\sigma_1 = 1$);

2. Type Square is the random set of points normally distributed relation to the square surface with the side $L = 0.5$ (expected mean $M_2 = 0$, root-mean-square deviation $\sigma_2 = 0,5$);

Table 1 and Table 2 some results of numerical simulation of convex hull are provided by means of the general recurrent algebraic method [Tyulyukin93, Shorikov97] and its modification. These are the indicators of the speed estimates: computation time and the number of the simplex algorithm iterations.

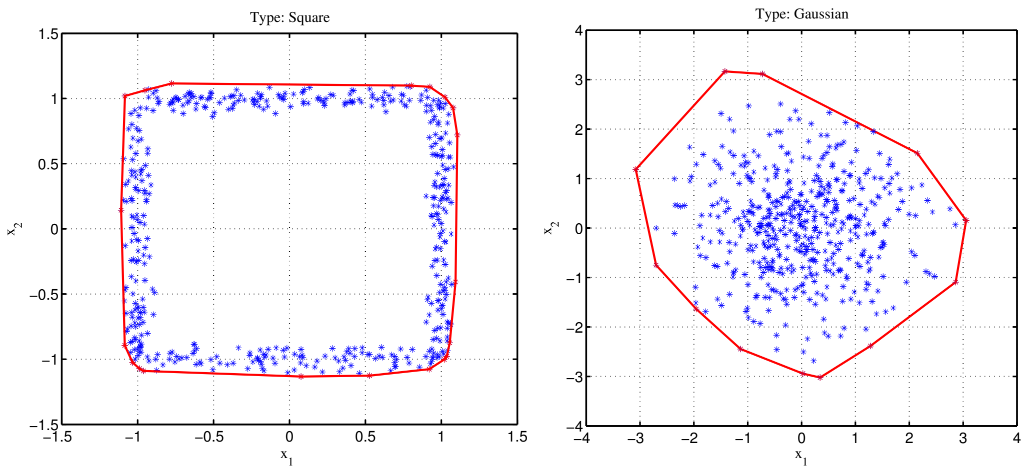

Figure 2: Convex hulls

Table 1: Results of numerical simulation for convex hull of reachable set computation problem on randomized points with Gaussian distribution law

Parameters		Computing time, s		Simplex algorithm iteration count		Number of vertices
Space dimension	Number of points	Recurrent algorithm	Modified algorithm	Recurrent algorithm	Modified algorithm	
n = 3	100	0,033	0,033	428	548	18
	500	0,167	0,097	2120	2360	29
	2000	1,053	0,317	8257	8542	44
	5000	5,132	0,732	20385	21021	54
	7000	7,897	1,548	28504	28744	60
n = 5	100	0,058	0,073	757	1114	52
	500	0,285	0,244	3960	5612	133
	2000	2,145	0,842	14201	16535	200
	5000	10,341	2,208	35403	39054	292
	7000	14,361	3,613	48280	51294	323
n = 7	100	0,262	0,381	1019	1649	82
	500	0,378	0,948	6055	9352	251
	2000	3,409	2,307	24804	34592	602
	5000	15,974	5,937	57758	73620	863
	7000	24,748	10,730	90783	95982	1011

Figure 3 graphically presents the performance of modified algorithm in comparison with general recurrent algorithm. Figure 3 shows that number of iterations is linearly depends on the number of points in the given set. However, the time required to solve convex hull problem grows nonlinearly. Therefore, with increasing of state space dimension of the system and time steps, the computing of reachable set of this system becomes inefficient in terms of computing time.

From the given results it is clear that modification of the general recurrent algebraic method due to sorting of an initial set, solving of smaller dimensions linear mathematical programming problems, allows to significantly reduce calculations time. Especially, it is noticeable for high dimensionality of Euclidean vector spaces.

Table 2: Results of numerical simulation for convex hull of reachable set computation problem on randomized points with "Square" law

Parameters		Computing time, s		Simplex algorithm iteration count		Vertices number
Space dimension	Points number	Recurrent algorithm	Modified algorithm	Recurrent algorithm	Modified algorithm	
n = 3	100	0,053	0,061	521	713	46
	500	0,230	0,164	2436	2837	85
	2000	1,146	0,517	9011	9400	94
	5000	3,863	1,202	21407	21951	96
	7000	6,850	1,936	30153	30622	117
n = 5	100	0,058	0,085	801	1333	91
	500	0,488	0,461	4705	6885	287
	2000	2,707	2,243	19584	25515	645
	5000	9,489	5,321	44837	53194	841
	7000	17,652	8,529	63787	74836	1116
n = 7	100	0,069	0,089	898	1605	100
	500	0,657	0,829	6491	11247	455
	2000	4,589	5,655	31522	49735	1421
	5000	21,691	19,115	81060	118841	2656
	7000	41,290	32,711	116654	165241	3345

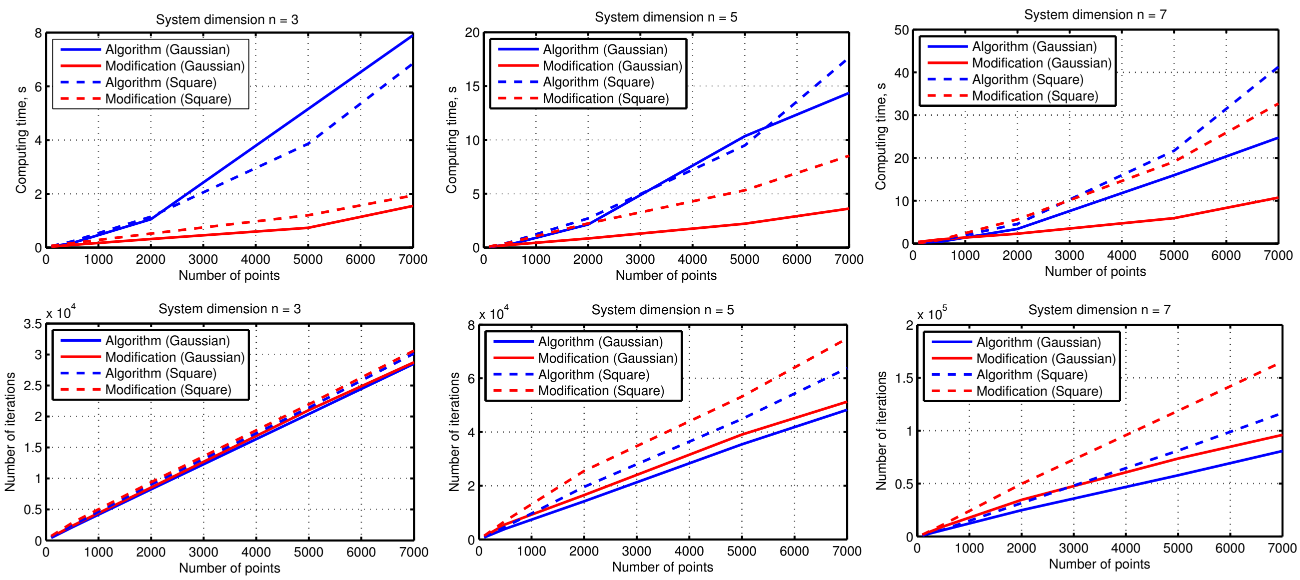

Figure 3: Numerical simulation results for convex hull of reachable set computation problem on randomized points

7 Conclusion

In this paper, we described modification of the general recurrent algebraic method for reachable set computation of linear discrete-time dynamical systems [Tyulyukin93, Shorikov97]. This modification is based on use of the ideas of "divide-and-conquer" algorithm [Pardalos95, Preparata85].

Also we provided the numerical simulation of exact reachable set computation for the dynamical systems of the 3rd order, described by linear discrete-time models. Let us note that the provided sorting algorithm is very effective. The best candidates for set extreme points are checked primarily, and the size of a set remains rather small throughout all computing process. As a result of application of the presented modification of a general recurrent algebraic method the goal was achieved and time spent for computing transactions was considerably reduced.

Program implementation of reachable set computation is performed by authors in the software environment MATLAB 8.4 (R2014a), where the modified recurrent algorithm of linear discrete-time dynamical systems was developed.

8 Acknowledgements

The research was supported by Russian Foundation for Basic Research (Project 15-01-02368).

References

[Chernikov68] S.N. Chernikov. *Linear Inequalities.* Moscow, Nauka Publ., 1968. (in Russian)

[Chernousko94] F.L. Chernousko. *State Estimation for Dynamic Systems.* Boca Raton: CRC Press, 1994.

[Girard05] A. Girard. Reachability of Uncertain Linear Systems Using Zonotopes. *Hybrid Systems: Computation and Control*, 3414:291-305, 2005.

[Krasovskii68] N.N. Krasovskii. *The Theory of Control of Motion (Linear Systems).* Moscow, Nauka Publ., 1968. (in Russian)

[Kurzhanski02] A.B. Kurzhanski, P. Varaiya. On ellipsoidal techniques for reachability analysis. *Optimization Methods and Software*, 17:177-207, 2002.

[Kurzhanskiy11] A.B. Kurzhanski, P. Varaiya. Reach set computation and control synthesis for discrete-time dynamical systems with disturbances. *Automatica*, 47:1414-1426, 2011.

[Lasserre91] J.B. Lasserre. Reachable and Controllable Sets for Two-Dimensional, Linear, Discrete-Time Systems. *Journal of Optimization Theory and Applications*, 70:583-595, 1991.

[Papadimitriou85] C.H. Papadimitriou, K. Steiglitz. *Combinatorial optimization: algorithms and complexity.* Moscow, Mir, 1985. (in Russian)

[Pardalos95] P.M. Pardalos, Y. Li, W.W. Hager. Linear Programming Approaches to the Convex Hull Problem in $\mathbb{R}^m$. *Computers & Mathematics with Applications*, 29(7):23-29, 1995.

[Preparata85] F. Preparata, Y. Li, M. Shamos. *Computational Geometry: An Introduction.* Springer-Verlag, 1985.

[Tyulyukin93] V.A. Tyulyukin, A.F. Shorikov. Algorithm for Handling the Terminal Control Problem for Linear Discrete System. *Automation and Remote Control*, 4:115-127, 1993.

[Shorikov97] A.F. Shorikov. *Minimax Estimation and Control in Discrete Dynamic Systems.* Yekaterinburg, Ural. Univers., 1997.

[Yudin69] D.B. Yudin, E.G. Goldstein. *Linear Programming.* Moscow, Nauka, 1969. (in Russian)

[Ziegler98] G.M. Ziegler. *Lectures on Polytopes.* Springer Verlag, New York, 1998.