UDC 519.85::[517+511]

M. Mishchenko

STRUCTURE-BASED OPTIMIZATION PROBLEM FOR MODEL PREDICTIVE CONTROL IN LINEAR MULTI-INPUT MULTI-VALUE SYSTEMS

Mykhailo Mishchenko

Institute for Applied System Analysis, National Technical University of Ukraine «Igor Sikorsky Kyiv Politechnic Institute», Kyiv,
mdmisch@firemail.cc, mdmisch@protonmail.com

Various technical and other real-world systems can be modelled with decent precision as linear systems. This approach is the core of the long established control theory, whose mathematical apparatus is ubiquitous when it comes to controlling some kind of system. While it is hard to underestimate importance of this approach, long history of research in this field showed some of its shortcomings which may hinder its application in various ways. For example, it does not allow to incorporate constraints on control signal's magnitude into the system's model. Thus, engineers are forced to manually tune controller's parameters ad-hoc in order to satisfy these constraints. This paper is dedicated to development of an alternative control algorithm based on the model predictive control approach. Its core idea is to generate control sequences by solving an optimization problem which objective function depends on predicted future state. It allows to generate fast stabilization trajectories without additional tuning by using the classic linear system's evolutionary equation as a future state predictor and constraints on controls as optimization problem's constraints. Meaningfully defined objective function is crucial in order to make this control algorithm work properly. It appeared that defining an objective function with good enough properties in general case is not a trivial task. This paper leverages modern nonstandard analysis in order to achieve this feat.

Keywords: MIMV system, linear system, optimization, model predictive control, stabilization, nonstandard analysis, hyperreals, hyperreal numbers, hyperreal-valued objective function.

It is a common engineering problem to control some kind of system in order to ensure its stability and prevent its malfunction or even self-destruction in case of reaching extreme states it can not sustain. These are so-called stabilization problems.

A wide range of systems which require such control are representable as linear multi-input multi-value (MIMV) systems, i. e. systems with n-dimensional real-valued state evolving in discrete time according to equation

$$x_{k+1} = Ax_k + Bu_k \tag{1}$$

and possible controls constrained with

$$u_k \in \mathcal{C} \subset \mathbb{R}^r, \tag{2}$$

where $\mathcal{C}$ is a closed bounded convex set.

Within the framework of this formalization, the stabilization problem can be loosely described as "how to bring the system from initial state x_0 to 0 with synthesized control sequence in minimum number of steps while preserving computational resources".

The classic approach to the stabilization problem, which originates from the control theory, is to create a feedback loop like

$$u_k = Dx_k. \tag{3}$$

 ISSN 2786-6491

This approach, as well as many other, share the same drawback: they do not use available control resources to their full potential. They do not stabilize the controlled system in the fastest possible way. For stable systems it is bearable as long as the method can guarantee that stabilization will be significantly faster than without any control at all. At the same time, for unstable systems (with spectral radius $\rho(A) > 1$) this becomes a problem, because constraints on possible control signals effectively limit the set of initial states x_0, for which it is a priori possible stabilize the system.

For unstable systems inefficient usage of control resources means that for a particular unstable system (1) with constraints (2) the set of initial states x_0 stabilizable by *a particular algorithm* is *just a subset* of all initial states x_0 stabilizable *a priori*. In practice, it is relatively easy to find for a particular unstable system (1) with constraints (2) a particular initial state x_0 stabilizable by one algorithm and not stabilizable by another (or even by the same algorithm with different metaparameters).

A classic workaround used to work with unstable systems is to say that a controlled system (1) combined with a controller (3) is a new system

$$x_{k+1} = (A + BD)\, x_k, \tag{4}$$

and to choose such D, that will make this combined system stable. But this approach just hides the fact that there are still some control signals somewhere inside of this combined system, and these control signals are still constrained.

The problem of inefficient usage of control resources becomes even more pronounced if any kind of random perturbations (noise) or uncertainties are added into the equation. Under such conditions if the system's state is near the stabilizable area's boundary, it can be thrown outside of it by random perturbations on first few steps, even though in deterministic case it would be theoretically possible to stabilize the system. Detrimental effect of random perturbations and uncertainties is the most dangerous when the system's state is near its stabilizable area's boundary. At the same time their influence is negligible when the system's state is near 0 (of course, if there is a controller which compensates it). This is why it is important to choose the most fast stabilization trajectory — in order to minimize timeframe in which the stabilization process can be affected by randomness and uncertainties in an irreversible way.

Thus, the general task in these regards is to define reasonably the most efficient way to utilize available control resources in terms of stabilization speed and formulate corresponding stabilization algorithm.

Considering the aforementioned drawbacks of the classic approach to this problem, it is natural to search for ways to mend them. Thus, the model predictive control (MPC) approach was applied [1–3] as an alternative.

The core idea of this approach is to explicitly find "the best" possible control sequence. In order to do this we compare different possible control sequences in terms of some kind of utility function. This way, "the best" control sequence is the one which has the lowest value of the utility function. Thus, to find this control sequence it is enough to solve a corresponding mathematical optimization problem.

The main caveat is that it is not practically feasible to optimize controls on infinite future horizon. Instead, we have to decide on a specific number of future steps (i. e. to decide on a horizon), for which controls are optimized.

Considering that our aim is to stabilize the system, or, in other words, to bring its state vector to zero at some point in future, it is reasonable to define the utility function as some kind of measure of future system state's closeness to zero. In particular, of the future state at the end of the prediction horizon, predicted from the sequence of controls on this horizon.

Computational experiments demonstrated [3], that while this approach works, in many cases it shows undesired behaviour. It is obvious, that for different initial states x_0 different number of steps is required to stabilize the system. If it happened that the prediction horizon length is equal to this minimum number of steps, the solution of the optimization problem is indeed the best possible control sequence. But if the horizon is longer, the computed solution would almost certainly stabilize the system exactly at the end of the prediction horizon and not earlier. And if it is shorter, the stabilization trajectory in most cases would differ from the initial part of any of the shortest stabilization trajectories, thus decreasing stabilization speed.

This misbehaviours force us to solve multiple optimization problems for different prediction horizon lengths in order to obtain the fastest stabilization trajectory. It drastically increases required computational resources, which becomes even more pronounced if we consider the fact that the problem's complexity also grows with increase of the prediction horizon's length.

Another unfortunate consequence is that the result of such procedure is a pregenerated control sequence for all duration of the predicted stabilization trajectory, which in most cases would be long. If the system is affected by random perturbations (noise) or if measurement of the current state is not precise, then real trajectory of the system would deviate from the predicted and thus this control sequence would become obsolete after first few steps. That is why we would essentially need to repeat this extensively complex computational procedure after each step of system's evolution, which increases computational burden even more. And it becomes even worse if we need to generate next control signals in real time.

All this could be avoided if the utility function (the notion of state's distance from zero) provided us close-to-ideal trajectories even on short prediction horizons. Experiments in [3] demonstrated, that combinations of systems and corresponding utility functions with such properties indeed exist. In particular, it was demonstrated that straightforward future state's Euclidean norm and prediction horizon equal to one gives trajectories identical (or nearly identical) to ideal ones for systems with diagonal matrix A in (1). This leads to a hypothesis, that for each particular system's structure there is a such well-behaving utility function. Thus, this work is devoted to finding a way of such utility functions' construction.

1. Problem statement

Let's begin with formal definition of the stabilization problem from the MPC point of view. The system (1), (2) has n dimensional state x evolving in time, so let there be some (at this point arbitrary) function $\mathcal{V} : \mathbb{R}^n \to [0, +\inf)$ which is intended to be used as a measurement of state's distance from 0, with following properties. It is:

- convex (i. e. $\forall a, b \in \mathbb{R}^n \ \forall r \in [0,1] \ \ \mathcal{V}\big(a + r(b-a)\big) \leq (1-r)\mathcal{V}(a) + r\mathcal{V}(b)$);
- has (global) minimum in 0;
- $\mathcal{V}(0) = 0$;
- $\forall x \neq 0 \ \mathcal{V}(x) > 0$.

Thus, if we want to stabilize the system (1), (2), then we want to bring future system's state as close as possible to 0, where "closeness" is expressed in terms of this function $\mathcal{V}$. If the current point of (discrete) time is k and the prediction horizon length we have choosen is s, then this our wish can be formally expressed as

$$\mathcal{V}\big(x_{k+s}(A, B, x_k, u_k, \ldots, u_{k+s-1})\big) \longrightarrow \min \tag{5}$$

$$u_k, \ldots, u_{k+s-1} \in \mathcal{C}, \tag{6}$$

where $x_{k+s}(\ldots)$ is a predicted future state, dependent on future controls $u_k, \ldots, u_{k+s-1}$, current system's state x_k and system's structure defined by matrices A and B.

Future state's prediction $x_{k+s}(\dots)$ can be easily obtained from (1), and it is as follows:

$$x_{k+s}(A, B, x_k, u_k, \dots, u_{k+s-1}) = \\ = A^s x_k + \sum_{i=0}^{s-1} A^{s-1-i} B u_{k+i}. \tag{7}$$

Thus, the stabilization problem (5), (6) is transformed into

$$\mathcal{V}\big(A^s x_k + \sum_{i=0}^{s-1} A^{s-1-i} B u_{k+i}\big) \longrightarrow \min \tag{8}$$

$$u_k, \dots, u_{k+s-1} \in \mathcal{C}. \tag{9}$$

The only and the main question left is how to define such function $\mathcal{V}(\cdot)$ with which the problem (8), (9) would produce close-to-ideal stabilization trajectories even on short prediction horizons. It is the topic of the following sections.

2. Optimal stabilization in systems with diagonalizable transition matrix A

Let's begin with a situation for which it is simple to derermine reasonably efficient objective function. If matrix A in (1) is diagonalizable, i. e. there is such matrix P, that A can be expressed as

$$A = P^{-1} \begin{pmatrix} \lambda_1 & 0 & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & \lambda_n \end{pmatrix} P, \tag{10}$$

then it would be beneficial to transform the state-space as in

$$y_k = P x_k. \tag{11}$$

In this transformed state-space the evolution equation (1) can be rewritten as

$$P^{-1} y_{k+1} = A P^{-1} y_k + B u_k \tag{12}$$

or, equivalently,

$$y_{k+1} = P A P^{-1} y_k + P B u_k. \tag{13}$$

From (10), (13) it is obvious that every component of y evolves independently from each other, if we do not consider the (arbitrary) way in which control vectors u are generated. Thus, it also becomes obvious, that the best possible control sequences $u_k, \dots, u_{k+s-1}$ are the ones, for which the PBu component pushes components of y to be the closest to 0. Or, more formally:

$$\begin{aligned} &\|y_{k+s}\| \longrightarrow \min \\ &u_k, \dots, u_{k+s-1} \in \mathcal{C} \subset \mathbb{R}^r, \end{aligned} \tag{14}$$

$$\begin{aligned} &\left\| P A^s P^{-1} y_k + \sum_{i=0}^{s-1} \left(P A^{s-1-i} P^{-1}\right) P B u_{k+i} \right\| \longrightarrow \min \\ &u_k, \dots, u_{k+s-1} \in \mathcal{C} \subset \mathbb{R}^r, \end{aligned} \tag{15}$$

$$\begin{aligned} &\left\| P\Big(A^s x_k + \sum_{i=0}^{s-1} A^{s-1-i} B u_{k+i}\Big) \right\| \longrightarrow \min \\ &u_k, \dots, u_{k+s-1} \in \mathcal{C} \subset \mathbb{R}^r, \end{aligned} \tag{16}$$

where $\|\cdot\|$ is a norm. In the following sections the Euclidean norm ($\|y\|_2 = \sqrt[2]{\sum_{i=1}^{n} y_i^2}$) will be used, because other norms may arise complications discussed further (and those which are fine still do not improve anything much).

By the way, we can control proirity of stabilization for different components of the transformed state-space by choosing a particular matrix P in transformation (10). If we have a particular transformation matrix P, we can reprioritize stabilization speed of different components of y with coefficients $\upsilon_1, \ldots, \upsilon_n > 0$ by choosing different transformation matrix $P' = \Upsilon P$, where

$$\Upsilon = \begin{pmatrix} \upsilon_1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \upsilon_n \end{pmatrix}. \tag{17}$$

As we will see below, the coefficients $\upsilon_1, \ldots, \upsilon_n$ can be choosen not only from the real line $\mathbb{R}$, but also from the hyperreal line $\mathbb{HR}$. This can give us some benefits even in this simple case, because it gives us more flexibility in prioritization.

3. General problem decomposition

In case when matrix A is not diagonalizable, it becomes much less obvious which future system state $x_{k+s}(x_k, A, B, u_k, \ldots, u_{k+s-1})$ is the best among possible ones (considering constraint (6)). Luckily, all possible pathological cases can be properly described in a systematic way in terms of the real-valued Jordan decomposition.

As we know, every possible (real-valued) matrix A can be decomposed into a block-diagonal form with four following types of blocks: 1×1 matrix (a scalar), rotation cell

$$R(a,b) = \begin{pmatrix} a & b \\ -b & a \end{pmatrix}, \quad a, b \in \mathbb{R} \setminus \{0\}, \tag{18}$$

aperiodic cell

$$L(\lambda) = \begin{pmatrix} \lambda & 1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & 1 \\ 0 & \cdots & \cdots & 0 & \lambda \end{pmatrix}, \quad \lambda \in \mathbb{R} \setminus \{0\}, \tag{19}$$

and rotation-aperiodic cell

$$S(a,b) = \begin{pmatrix} \begin{array}{cc|cc|c} a & b & 1 & 0 & \\ -b & a & 0 & 1 & \\ \hline & & a & b & \ddots \\ & & -b & a & \\ \hline & & & & \ddots \end{array} & & \\ & \ddots & \\ & & \begin{array}{cc|cc} a & b & 1 & 0 \\ -b & a & 0 & 1 \\ \hline & & a & b \\ & & -b & a \end{array} \end{pmatrix}, \quad a, b \in \mathbb{R} \setminus \{0\}. \tag{20}$$

As with example in the previous section, each of resulting cells naturally corresponds to a linear subspace of the original state-space, projection of the state x on which evolves independently from projections on linear subspaces corrresponding to other cells. It means

ISSN 2786-6491

that impact of our controls (and of possible random perturbations) on future system's states remains isolated inside each corresponding linear subspace. In other words, consequences of projection on such linear subspace of control impact Bu remain isolated inside the same linear subspace indefinitely.

So, let there be a Jordan decomposition

$$A = P^{-1}JP \tag{21}$$

of the feedback matrix A.

What we can immediately see in aperiodic cells (19), (20) is that all of their corresponding components of Px_k except for one impact not only their own future values in Px_{k+1}, but also of neighboring ones. Thus, in order to be able to use the same approach, as with diagonalizable feedback matrices A, it is important to reduce this unwanted impact as much as possible.

Theoretically, we can do it as in

$$L(\lambda) = \begin{pmatrix} \varepsilon^{-(n-1)} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 \end{pmatrix} \begin{pmatrix} \lambda & \varepsilon & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ & & \ddots & \ddots & \varepsilon \\ 0 & \cdots & \cdots & 0 & \lambda \end{pmatrix} \begin{pmatrix} \varepsilon^{n-1} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 \end{pmatrix}, \tag{22}$$

$$\begin{aligned} S(a,b) = {} & \left(\begin{array}{cc|c} \varepsilon^{-(n-1)} & 0 & \\ 0 & \varepsilon^{-(n-1)} & \\ \hline & & \ddots \\ & & \end{array}\right. \\ & \left.\begin{array}{c|cc} & & \\ \hline \ddots & & \\ \hline & \varepsilon^{-0} & 0 \\ & 0 & \varepsilon^{-0} \end{array}\right) \times \\ & \times \left(\begin{array}{cc|cc|c} a & b & \varepsilon & 0 & \\ -b & a & 0 & \varepsilon & \\ \hline & & a & b & \\ & & -b & a & \\ \hline & & & & \ddots \\ & & & & \end{array}\right. \\ & \left.\begin{array}{c|cc|cc} & & & & \\ \hline \ddots & & & & \\ \hline & a & b & \varepsilon & 0 \\ & -b & a & 0 & \varepsilon \\ \hline & & & a & b \\ & & & -b & a \end{array}\right) \times \\ & \times \left(\begin{array}{cc|c} \varepsilon^{n-1} & 0 & \\ 0 & \varepsilon^{n-1} & \\ \hline & & \ddots \\ & & \end{array}\right. \\ & \left.\begin{array}{c|cc} & & \\ \hline \ddots & & \\ \hline & \varepsilon^{0} & 0 \\ & 0 & \varepsilon^{0} \end{array}\right), \end{aligned} \tag{23}$$

where the smaller $\varepsilon > 0$ we choose, the more independent become dimensions in the feedback loop. The drawback of this approach is that in practice by decreasing ε we would quickly encounter numerical problems in computations while calculating an optimal control on a computer. Thus, to achieve better results we need to extend the notion of real

numbers we ubiquitously use in a way that would allow us to consistently operate with concept of an infinitesimal number.

As a side note, you may notice that rotation Jordan blocks in form (18) are not diagonalizable (if we are not using complex numbers) and so we can not untangle them by state-space transformations even to some degree. The same also applies to the rotation-aperiodic cells (20): we can untangle from each other different rotation cells they are consisting of as in (23), but we can not untangle them completely. But, considering their nature, it is still fine: they produce a rotating vector in a projection of the state onto the corresponding linear 2-d subspace, and thus, considering that our aim is system's stabilization, we need to minimize its *Euclidean norm*, but not the independent corresponding transformed state-space components. If we use the Euclidean norm as a measure of whole state's distance from zero in (14)–(16), then this caveat does not make any difference. But if we want for some reason to use another kind of norm, then we must make sure, that this norm of our choice is calculated not from individual state vector's components, but from:

• components corresponding to scalar (1×1) Jordan cells;

• components corresponding to untangled variants of aperiodic Jordan cells (19) (as in (22));

• Euclidean norms of component pairs corresponding to rotation Jordan cells (18);

• Euclidean norms of component pairs corresponding to rotation blocks of untangled variants of rotation-aperiodic Jordan cells (20) (as in (23)).

For example, let us suppose that we want to use a variant of the so-called p-norm not identical to the Euclidean norm ($\|v\|_p = \sqrt[p]{\sum_{i=1}^{n} v_i^p},\ p \in [1,2) \cup (2,+\infty)$). In this case if we have untangled Jordan decomposition

$$\left(\begin{array}{cc|cc|cc|cc|c}
1/2 & \varepsilon_1 & & & & & & & 1 \\
0 & 1/2 & & & & & & & 2 \\
\hline
 & & 2 & & & & & & 3 \\
\hline
 & & & 1 & 1 & & & & 4 \\
 & & & -1 & 1 & & & & 5 \\
\hline
 & & & & & \sqrt{2} & \sqrt{2} & \varepsilon_2 & 0 & 6 \\
 & & & & & -\sqrt{2} & \sqrt{2} & 0 & \varepsilon_2 & 7 \\
\hline
 & & & & & & & \sqrt{2} & \sqrt{2} & 8 \\
 & & & & & & & -\sqrt{2} & \sqrt{2} & 9
\end{array}\right), \tag{24}$$

we must use norm constructed as following:

$$\|y\| = \left\|(y_1, y_2, y_3, \|(y_4, y_5)^{\mathrm{T}}\|_2, \|(y_6, y_7)^{\mathrm{T}}\|_2, \|(y_8, y_9)^{\mathrm{T}}\|_2)^{\mathrm{T}}\right\|_p. \tag{25}$$

4. On the notion of the hyperreal line

The following sections will heavily use the notin of hyperreal numbers. As it is not among mainstream mathematical instruments, this section will discuss some of their properties which are of interest in this paper.

The most distinct feature of the hyperreal number line $\mathbb{HR}$ is that, while containing real numbers in itself, it introduces infinitesimal numbers. A positive infinitesimal is such number d, that it is bigger than 0 and smaller than every positive real number ($(d > 0) \wedge \forall \varepsilon \in \mathbb{R}\,(\varepsilon > 0) \to (d < \varepsilon)$); a negative infinitesimal is such number d, that it is smaller than 0 and bigger than every negative real number ($(d < 0) \wedge \forall \varepsilon \in \mathbb{R}\,(\varepsilon < 0) \to (d > \varepsilon)$).

Considering that all arithmetic operations and functions defined for the real line are also applicable for the hyperreals, this automatically produces quite rich set of nonstandard numbers. For example, if there exists some positive infinitesimal d, then there is also $-d$ (which is an example of negative infinitesimal), all of its powers, all of their linear combinations and so on. And, for instance, its integer powers satisfy

$$0 < \ldots < d^3 < d^2 < d < \overset{\parallel}{d^0} < d^{-1} < d^{-2} < d^{-3} < \ldots \quad , \tag{26}$$

where its negative powers (i. e. d^{-1}) are examples of positive infinite hyperreals (which are bigger than any real number).

Having the notion of infinitesimal numbers, the notion of infinitesimally close numbers naturally arises. Two hyperreals $a, b \in \mathbb{HR}$ are infinitesimally close ($a \approx b$) if $a - b$ is infinitesimal. And so, there is another valuable property: any finite hyperreal is infinitely close to some real number, which is called its *shadow*. For example, shadow of infinitesimal number d is $\boxed{d} = 0$, and $\boxed{1 + d^2} = \boxed{1} + \boxed{d^2} = 1$. Infinite hyperreals, obviously, do not have shadows.

Every set of real numbers $\mathcal{S} \subset \mathbb{R}$ has its "enlarged" hyperreal counterpart ${}^*\mathcal{S} \subset \mathbb{HR}$. This way, for example, a real line segment $[a, b] \coloneqq \{x \in \mathbb{R} : (x \geq a) \wedge (x \leq b)\}$ has its enlarged counterpart ${}^*[a, b] = \{x \in \mathbb{HR} : (x \geq a) \wedge (x \leq b)\}$, which contains:

- all real numbers in $[a, b]$;
- all hyperreals infinitely close to real numbers in (a, b);
- all hyperreals infinitely close to *a from right*;
- all hyperreals infinitely close to *b from left*.

Theorem (Robinson's Compactness Criterion)**.** *$\mathcal{S}$ is a compact set in $\mathbb{R}^n$ iff for every $l \in {}^*\mathcal{S}$ exists $\boxed{l} \in \mathcal{S}$* [4, p. 117].

Naturally, enlargement of any finite set of real numbers is the set itself (${}^*\{x_1, \ldots, x_n\} = \{x_1, \ldots, x_n\},\ n \in \mathbb{N}$) and enlargement of the whole real line is the hyperreal line (${}^*\mathbb{R} = \mathbb{HR}$).

Finally, the most precious property of the hyperreal line is the *transfer principle*. It can be loosely described as following: every true statement about real numbers formally written in the first order logic has its counterpart about hyperreal numbers with all sets replaced by their enlargements, which is also true. This is the reason why hyperreals inherit many valuable properties of real numbers, such as the fact that the hyperreal line is also an ordered field.

Nevertheless, when using the hyperreal calculus it is important to be careful with applying well-known statements about real numbers onto hyperreal numbers. Being able to do such transfer is one of the most important of their properties. But there is a caveat: not all possible statements can be transferred. It is due to the fact that not all statements can be described as a first order logic formula. For example, the statement that every upper-bounded set of numbers has a least upper bound is true in $\mathbb{R}$, but is not true in $\mathbb{HR}$. It is trivial to see, that set of all finite hyperreals is upper-bounded by any positive infinite hyperreal, but its least upper bound does not exist ([4, p. 11], [5, Chapter 4]).

If reader is interested in more deep understanding of construction and usage of hyperreal numbers, here are works the author of this paper consulted with while preparing it. For detailed and explicit construction of (different variants of) hyperreal line and for formal discussion of their properties see [6]. For reasonably complete alternative calculus built on top of hyperreals see [7]. For lightweight informal introduction into the notion of hyperreals and the role of mathematical logic in describing their properties, see [5]. For a complete lecture course on hyperreals adapted to a textbook, see [4].

5. Defining optimal control with hyperreal-valued objective function

Let's again consider equation (21). As we know, the Jordan decomposition matrix J can be represented as

$$J = \begin{pmatrix} \Psi_1 & & & \\ & \Psi_2 & & \\ & & \ddots & \\ & & & \Psi_m \end{pmatrix}, \tag{27}$$

where each of $\Psi_1, \ldots, \Psi_m$ is either a scalar, or one of (18)–(20).

Thus, we can further decompose each cell as $\Psi_i = \Delta_i^{-1}\Lambda_i\Delta_i,\ i \in \{1, \ldots, m\}$, where:

• $\Delta_i = \delta_i \in \{\delta' \in \mathbb{HR} \mid \delta' > 0\}, \Lambda_i = \Psi_i$, if $\Psi_i \in \mathbb{R}$;

• $\Delta_i = \begin{pmatrix} \delta_i & 0 \\ 0 & \delta_i \end{pmatrix}$ for some $\delta_i \in \{\delta' \in \mathbb{HR} \mid \delta' > 0\}, \Lambda_i = \Psi_i$, if $\Psi_i = R(a, b)$ for some $a, b \in \mathbb{R} \setminus \{0\}$ (see (18));

• $\Delta_i = \begin{pmatrix} \delta_{i,n_i} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \delta_{i,1} \end{pmatrix}$, $\Lambda_i = \begin{pmatrix} \lambda & \frac{\delta_{i,n_i-1}}{\delta_{i,n_i}} & 0 & \cdots & 0 \\ & \ddots & \ddots & \ddots & \vdots \\ 0 & & \ddots & \ddots & 0 \\ \vdots & & & \ddots & \frac{\delta_{i,1}}{\delta_{i,2}} \\ 0 & \cdots & \cdots & 0 & \lambda \end{pmatrix}$

for some $\delta_{i,1}, \ldots, \delta_{i,n_i} \in \{\delta' \in \mathbb{HR} \mid \delta' > 0\}$,
where $\frac{\delta_{i,j}}{\delta_{i,j+1}} \approx 0$ for $j \in 1, \ldots, (n_i - 1)$,
if $\Psi_i = L(\lambda)$ for some $\lambda \in \mathbb{R}$ (see (19));

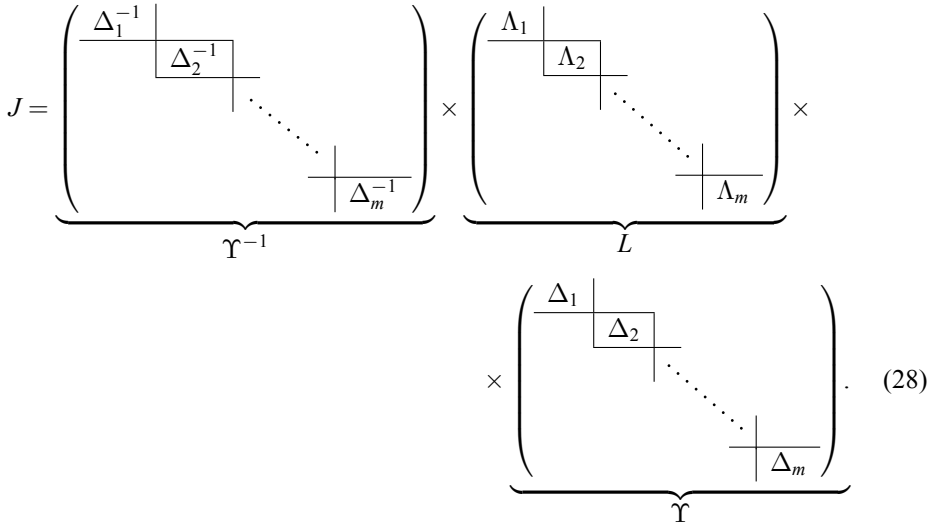
• $\Delta_i = \begin{pmatrix} \begin{matrix} \delta_{i,n_i} & 0 \\ 0 & \delta_{i,n_i} \end{matrix} & & & \\ & \ddots & & \\ & & \begin{matrix} \delta_{i,1} & 0 \\ 0 & \delta_{i,1} \end{matrix} \end{pmatrix}$,

$\Lambda_i = \begin{pmatrix} \begin{matrix} a & b \\ -b & a \end{matrix} & \begin{matrix} \frac{\delta_{i,n_i-1}}{\delta_{i,n_i}} & 0 \\ 0 & \frac{\delta_{i,n_i-1}}{\delta_{i,n_i}} \end{matrix} & & & \\ & \begin{matrix} a & b \\ -b & a \end{matrix} & \ddots & & \\ & & \ddots & \begin{matrix} a & b \\ -b & a \end{matrix} & \begin{matrix} \frac{\delta_{i,1}}{\delta_{i,2}} & 0 \\ 0 & \frac{\delta_{i,1}}{\delta_{i,2}} \end{matrix} \\ & & & & \begin{matrix} a & b \\ -b & a \end{matrix} \end{pmatrix}$

for some $\delta_{i,1}, \ldots, \delta_{i,n_i} \in \{\delta' \in \mathbb{HR} \mid \delta' > 0\}$,
where $\frac{\delta_{i,j}}{\delta_{i,j+1}} \approx 0$ for $j \in 1, \ldots, (n_i - 1)$,
if $\Psi_i = S(a, b)$ for some $a, b \in \mathbb{R} \setminus \{0\}$ (see (20)).

 ISSN 2786-6491

This way we can say that



(28)

So, from (21) we can produce

$$A = P^{-1}\Upsilon^{-1}L\Upsilon P. \tag{29}$$

As we can see, the shadow $\boxed{L}$ of matrix L is a block-diagonal matrix with only scalar (1×1) and rotation (18) blocks. Thus, we can obtain such state-space transformation $y = \Upsilon P x$, whose components corresponding to 1×1 blocks and pairs of components corresponding to the rotation blocks (18) evolve *almost independently* of each other. It allows us to conveniently define the objective function as $\|y_{k+s}\|_2 \to \min$ in the same way, as we have done it in Section 2. Thus, the convex optimization problem which is proposed to be used for control synthesis can be represented as

$$\underbrace{\left\|\Upsilon P\Big(A^s x_k + \sum_{i=0}^{s-1} A^{s-1-i} B u_{k+i}\Big)\right\|_2}_{\sqrt[2]{F_0(u_k,\ldots,u_{k+s-1})}} \longrightarrow \min \tag{30}$$

$$u_k, \ldots, u_{k+s-1} \in \mathcal{C} \subset \mathbb{R}^r. \tag{31}$$

Similarly to the example with diagonalizable matrix A, it allows the norm in the objective function to capture changes in the *almost-independent* components. This way it is able to capture long-term benefits or losses from different possible control sequences in more consistent and explicit way. It is a significatnt improvement if compared with straightforward minimization of future state's distance from zero $\|x_{k+s}\|$, because the latter is indifferent to the system's feedback loop structure defined by the matrix A. In particular, a lower value of $\|x_{k+s}\|$ can hide more detrimental future impact of a corresponding future state, than of some another possible state with higher value of the same norm.

The fact that coefficients on the main diagonal of Υ are hyperreals gives us another valuable instrument in controller design: now we can explicitly specify that one of the components of the transformed state-space has infinite times bigger stabilization priority than another. This is exactly what happens for coefficients of Λ corresponding to untangled aperiodic and rotational-aperiodic cells according to the decomposition above. But, aside from these rules, we can freely tune comparative priorities of transformed state's dimensions corresponding to different Jordan cells. For example, we can make priority of dimensions corresponding to unstable cells infinite times bigger then of those corresponding to stable ones.

It is also important to notice that while the objective function in (30), (31) is hyperreal-valued, its variables $u_k, \ldots, u_{k+s-1}$ are constrained by the set $\mathcal{C}$, which is a subset of $\mathbb{R}^r$ and thus is convex *only in* $\mathbb{R}^r$, but not in $\mathbb{HR}^r$. It is reasonable to narrow the search area of the problem's solution to the real-valued vector space because in practice it is not possible, for example, to apply a hyperreal-valued amount of voltage to some kind of electronic device.

6. On existence of the solution

Before trying to solve (some variant of) the problem (30), (31) it is important to discuss whether its optimum exists. While the Extreme Value Theorem is a well-known result in the classic mathematical analysis, in the nonstandard analysis, which arises from the notion of hyperreal numbers, we need to be careful with transferring seemingly obvious statements onto nonstandard entities. It is proven for one-dimensional case [4, p. 80], but for (30), (31) we need to prove this theorem in a quite specific formulation.

Theorem (Extreme Value Theorem)**.** *If there is a continuous function* $f\colon \mathbb{R}^m \times \mathcal{S} \to \mathbb{R}$*, where* $\mathcal{S}$ *is a compact nonempty subset of* $\mathbb{R}^n$*, then for each* $h \in \mathbb{HR}^m$ *exist* $g_{\max}(h) = {} = \arg\max_{g \in {}^*\mathcal{S}} {}^*f(h,g)$ *and* $g_{\min}(h) = \arg\min_{g \in {}^*\mathcal{S}} {}^*f(h,g)$.

Proof. It is a known fact, that every continuous real-valued function defined on any compact nonempty set $\mathcal{S} \in \mathbb{R}^n$ has its maximum and minimum values on it. The following can be stated as its corollary for every continuous function $f\colon \mathbb{R}^m \times \mathcal{S} \to \mathbb{R}$:

$$\forall h \in \mathbb{R}^m \; \exists g_{\max} \in \mathcal{S} \; \forall g \in \mathcal{S} \; f(h,g) \leq f(h, g_{\max}), \tag{32}$$

$$\forall h \in \mathbb{R}^m \; \exists g_{\min} \in \mathcal{S} \; \forall g \in \mathcal{S} \; f(h,g) \geq f(h, g_{\min}). \tag{33}$$

Thus, by the transfer principle:

$$\forall h \in \mathbb{HR}^m \; \exists g_{\max} \in {}^*\mathcal{S} \; \forall g \in {}^*\mathcal{S} \; f(h,g) \leq f(h, g_{\max}), \tag{34}$$

$$\forall h \in \mathbb{HR}^m \; \exists g_{\min} \in {}^*\mathcal{S} \; \forall g \in {}^*\mathcal{S} \; f(h,g) \geq f(h, g_{\min}). \tag{35}$$

□

Corollary. *Objective function* (30) *has its minimum and maximum on* ${}^*(\mathcal{C}^s)$.

It is important to notice, that in general case it is not guaranteed that $g_{\max}(h)$ and/or $g_{\min}(h)$ are real-valued. Moreover, there are trivial cases when they are indeed nonstandard (not real-valued). We also don't know whether there are any $f(\cdot,\cdot)$ and h for which $\min_{g \in \mathcal{S}} f(h,g)$ and/or $\max_{g \in \mathcal{S}} f(h,g)$ does not exist (while $\min_{g \in {}^*\mathcal{S}} f(h,g)$ and $\max_{g \in {}^*\mathcal{S}} f(h,g)$ still exists, as follows from this variant of the Extreme Value Theorem).

Nevertheless, the fact that (30), (31) always has real-valued solution will be shown explicitly in the following section.

7. Decomposition of the convex optimization problem with hyperreal-valued objective function

This section will discuss an explicit algorithm of solving the problem (30), (31) with hyperreal-valued objective function by solving a sequence of problems with real-valued objective function it is decomposed into. This procedure, described as in form of mathematical induction, will also serve as a proof of the problem solution's existence.

The idea of this procedure is to sequentially cut off parts of the set of possible controls (31) in such way, that it would each time decrease maximum value of the objective function (30) on the remaining subset of controls. As it will be demonstrated in the mathematical induction's stop condition, after finite number of steps the remaining part will produce equal values of the objective function (30), which thus are its minimum value on the set of possible controls.

7.1. Decomposition-based problem solving procedure.

Induction base. Let there be a diagonal hyperreal-valued matrix $\Upsilon_0 \coloneqq \Upsilon$ (obtained from the procedure described in the Section 5), an objective function

$$F_0(u_k, \ldots, u_{k+s-1}) \coloneqq \left\| \Upsilon_0 P\Big(A^s x_k + \sum_{i=0}^{s-1} A^{s-1-i} B u_{k+i}\Big) \right\|_2^2 \tag{36}$$

and its domain

$$\mathcal{S}_0 \coloneqq \mathcal{C}^s \subset \mathbb{R}^{r \cdot s}, \tag{37}$$

which are identical to (30) and (31) correspondingly.

Induction step. There is a diagonal matrix $\Upsilon_{q-1} = \operatorname{diag}(\upsilon_{q-1,1}, \ldots, \upsilon_{q-1,n})$, objective function

$$F_{q-1}(u_k, \ldots, u_{k+s-1}) \coloneqq \left\| \Upsilon_{q-1} P\Big(A^s x_k + \sum_{i=0}^{s-1} A^{s-1-i} B u_{k+i}\Big) \right\|_2^2 \tag{38}$$

and its domain $\mathcal{S}_{q-1} \subset \mathcal{C}^s \subset \mathbb{R}^{r \cdot s}$. Υ_{q-1} has all of its coefficients nonnegative and there is at least one nonzero coefficient among them. F_{q-1} preserves order with F_0 on $\mathcal{S}_{q-1}$ (i. e. for any $(u_k, \ldots, u_{k+s-1}) \in \mathcal{S}_{q-1}$ and $(u'_k, \ldots, u'_{k+s-1}) \in \mathcal{S}_{q-1}$ we have $F_{q-1}(u_k, \ldots, u_{k+s-1}) \leq$ $\leq F_{q-1}(u'_k, \ldots, u'_{k+s-1})$ iff $F_0(u_k, \ldots, u_{k+s-1}) \leq F_0(u'_k, \ldots, u'_{k+s-1})$).

From the Extreme Value Theorem follows, that (38) maps set of controls $\mathcal{S}_{q-1}$ onto an interval $\{f \in \mathbb{HR} : f_{\min} \leq f \leq f_{\max}\}$, where $f_{\min}$ and $f_{\max}$ are the minimum and maximum possible values of (38) on ${}^*(\mathcal{S}_{q-1}) \supset \mathcal{S}_{q-1}$.

Let us fix an index i_{q-1} such that $\upsilon_{q-1,i_{q-1}}$ is the biggest among dagonal coefficients of Υ_{q-1} (or one of the biggest, if there are several equal ones which are bigger than any other). Then, a diagonal matrix $\Upsilon'_{q-1} \coloneqq \left(\frac{1}{\upsilon_{q-1,i_{q-1}}} \Upsilon_{q-1}\right)$ has all of its coefficients in ${}^*[0,1]$, and at least one of them is equal to 1. Thus, there exists its shadow $\boxed{\Upsilon'_{q-1}}$.

Let there be a problem

$$\boxed{\left\| \Upsilon'_{q-1} P\Big(A^s x_k + \sum_{i=0}^{s-1} A^{s-1-i} B u_{k+i}\Big) \right\|_2^2} \longrightarrow \min, \tag{39}$$

$$(u_k, \ldots, u_{k+s-1}) \in \mathcal{S}_{q-1} \subset \mathcal{C}^s \subset \mathbb{R}^{r \cdot s}. \tag{40}$$

Considering that controls $u_k, \ldots, u_{k+s-1}$ are real-valued, we can write that

$$\boxed{\left\| \Upsilon'_{q-1} P\Big(A^s x_k + \sum_{i=0}^{s-1} A^{s-1-i} B u_{k+i}\Big) \right\|_2^2} =$$

$$= \left\| \boxed{\Upsilon'_{q-1}} P\Big(A^s x_k + \sum_{i=0}^{s-1} A^{s-1-i} B u_{k+i}\Big) \right\|_2^2. \tag{41}$$

Then, considering the way the set $\mathcal{S}_{q-1}$ was defined on the previous iteration, the problem (39), (40) becomes an example of classic convex optimization problem, which can be solved explicitly (for example, with the CVXOPT solver [8]). We denote set of its solutions as $\mathcal{S}_q \subset \mathcal{S}_{q-1} \subset \mathcal{C}^s$.

It is an obvious fact that if two hyperreals a, and b are finite, then from $\boxed{a} < \boxed{b}$ follows $a < b$. As $\mathcal{S}_q \subset \mathcal{S}_{q-1}$ is a set of solutions of (39), (40), we can conclude that for any combination of controls from $\mathcal{S}_q$ the value of the objective function F_{q-1} is strictly less than any other combination of controls from $\mathcal{S}_{q-1} \setminus \mathcal{S}_q$.

F_{q-1} preserves order with F_0 on $\mathcal{S}_{q-1}$, so, again, for any combination of controls from $\mathcal{S}_q$ the value of the objective function F_0 is strictly less than any other combination of controls from $\mathcal{S}_{q-1} \setminus \mathcal{S}_q$.

Considering that $\mathcal{S}_q \subset \mathcal{S}_{q-1} \subset \ldots \subset \mathcal{S}_0$ and that we have the same property for all previous steps, we conclude that for any combination of controls from $\mathcal{S}_q$ the value of the objective function F_0 is strictly less than any other combination of controls from $\mathcal{S}_0 \setminus \mathcal{S}_q$.

We know that on $\mathcal{S}_q$ the value of (39) is a constant, so let's denote this value as φ_{q-1}. Function F'_q defined as

$$F'_q(u_k,\ldots,u_{k+s-1}) \coloneqq \frac{1}{\upsilon_{q-1,i_{q-1}}} F_{q-1}(u_k,\ldots,u_{k+s-1}) - \\ - \min_{l:\ \boxed{\frac{\upsilon_{q-1,l}}{\upsilon_{q-1,i_{q-1}}}} \neq 0} \left(\frac{\left(\frac{\upsilon_{q-1,l}}{\upsilon_{q-1,i_{q-1}}}\right)}{\boxed{\frac{\upsilon_{q-1,l}}{\upsilon_{q-1,i_{q-1}}}}} \right)^2 \varphi_{q-1}. \quad (42)$$

preserves order with F_{q-1} on $\mathcal{S}_q$ (i. e. $F_{q-1}(u_k,\ldots,u_{k+s-1}) \leq F_{q-1}(u'_k,\ldots,u'_{k+s-1})$ iff $F'_q(u_k,\ldots,u_{k+s-1}) \leq F'_q(u'_k,\ldots,u'_{k+s-1})$), so it also preserves order with F_0 on $\mathcal{S}_q$.

Let us denote rows of P as $p_1,\ldots,p_n$. If we reduce domain of function F'_q to $\mathcal{S}_q$, we will obtain the following:

$$F_q(u_k,\ldots,u_{k+s-1}) \coloneqq F'_q\Big|_{\mathcal{S}_q}(u_k,\ldots,u_{k+s-1}) = \\ = \sum_{j=1}^{n} \left(\frac{\upsilon_{q-1,j}}{\upsilon_{q-1,i_{q-1}}} p_j\Big(A^s x_k + \sum_{i=0}^{s-1} A^{s-1-i} B u_{k+i}\Big) \right)^2 - \\ - \min_{l:\ \boxed{\frac{\upsilon_{q-1,l}}{\upsilon_{q-1,i_{q-1}}}} \neq 0} \left(\frac{\left(\frac{\upsilon_{q-1,l}}{\upsilon_{q-1,i_{q-1}}}\right)}{\boxed{\frac{\upsilon_{q-1,l}}{\upsilon_{q-1,i_{q-1}}}}} \right)^2 \times \\ \times \sum_{j=1}^{n} \left(\boxed{\frac{\upsilon_{q-1,j}}{\upsilon_{q-1,i_{q-1}}}} p_j\Big(A^s x_k + \sum_{i=0}^{s-1} A^{s-1-i} B u_{k+i}\Big) \right)^2 = \\ = \sum_{j=1}^{n} \underbrace{\left(\left(\frac{\upsilon_{q-1,j}}{\upsilon_{q-1,i_{q-1}}}\right)^2 - \min_{l:\ \boxed{\frac{\upsilon_{q-1,l}}{\upsilon_{q-1,i_{q-1}}}} \neq 0} \left(\frac{\left(\frac{\upsilon_{q-1,l}}{\upsilon_{q-1,i_{q-1}}}\right)}{\boxed{\frac{\upsilon_{q-1,l}}{\upsilon_{q-1,i_{q-1}}}}} \right)^2 \cdot \boxed{\frac{\upsilon_{q-1,j}}{\upsilon_{q-1,i_{q-1}}}}^2 \right)}_{\upsilon^2_{q,j}} \times \\ \times \left(p_j\Big(A^s x_k + \sum_{i=0}^{s-1} A^{s-1-i} B u_{k+i}\Big) \right)^2 = \\ = \sum_{j=1}^{n} \left(\upsilon_{q,j} p_j\Big(A^s x_k + \sum_{i=0}^{s-1} A^{s-1-i} B u_{k+i}\Big) \right)^2 = \\ = \left\| \Upsilon_q P\Big(A^s x_k + \sum_{i=0}^{s-1} A^{s-1-i} B u_{k+i}\Big) \right\|_2^2. \quad (43)$$

Stop condition. By construction, at each step q the next obtained diagonal matrix Υ_q has at least one more coefficient on the main diagonal equal to zero, if compared to the Υ_{q-1}. So, there is a step t at which we will obtain $\Upsilon_t = 0$. Thus, $F_t(u_k, \ldots, u_{k+s-1}) \equiv \equiv 0$. But F_t preserves order with F_0 on $\mathcal{S}_t$. So, F_0 has equal values on $\mathcal{S}_t$. And from the last step we have that for any $(u_k, \ldots, u_{k+s-1}) \in \mathcal{S}_t$ and any $(u'_k, \ldots, u'_{k+s-1}) \in \mathcal{S}_0 \setminus \mathcal{S}_t$ $F_0(u_k, \ldots, u_{k+s-1}) < F_0(u'_k, \ldots, u'_{k+s-1})$. Or, in other words, $\mathcal{S}_t$ is the (nonempty) set of solutions of the problem (30), (31).

7.2. On usage of non-Euclidean norms. While there seems to be no practical reason to use norms other than Euclidean, it is possible to use some of them. For instance, the procedure and proof in the previous subsection can be adapted to the example of the p-norm based composite norms constructed like (25) from Section 3. But there may be caveats with other norms.

First of all, we must make sure that the non-Euclidean norm we want to use satisfies precautions about rotation and rotation-aperiodic cells, described at the end of Section 3. But even if it satisfies them, it still may produce unwanted effects. For example, if we want to use the maximum norm $\|v\|_{\inf} = \max_{i=1}^{n} |v_i|$ as a basis for composite norm complying with the aforementioned precautions, it may (depending on coefficients of matrix Υ) leave some of the dimensions effectively uncontrolled until other ones are completely stabilized. Other nonstandard norms may even maximize part of components instead of minimizing them if there is a coefficient v_i infinitesimally smaller than other ones (i. e. if there is v_j such that $\frac{v_i}{v_j} \approx 0$) and $\frac{\partial \|y\|}{\partial y_i} \left((y_1, \ldots, y_{i-1}, 0, y_{i+1}, \ldots y_n)^{\mathrm{T}}\right) \neq 0$.

М.Д. Міщенко

СТРУКТУРНО-ОБУМОВЛЕНА ЗАДАЧА ОПТИМІЗАЦІЇ ДЛЯ КЕРУВАННЯ ЗА ПРОГНОЗНОЮ МОДЕЛЛЮ У ЛІНІЙНИХ СИСТЕМАХ ІЗ БАГАТЬМА ЗМІННИМИ ТА ВХОДАМИ

Михайло Дмитрович Міщенко

Інститут прикладного системного аналізу, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», м. Київ, *mdmisch@firemail.cc, mdmisch@protonmail.com*

Різноманітні технічні та інші системи можуть бути змодельовані із пристойною точністю як лінійні системи. Цей підхід є основою давно відомої і загальноприйнятої теорії керування, математичний апарат якої є невід'ємним, коли потрібно керувати деякою системою. Хоча важливість цього підходу важко недооцінити, тривала історія досліджень у цьому напрямку показала деякі його недоліки, які можуть у різні способи заважати його застосуванню. Наприклад, він не дозволяє включити обмеження на величину керуючого сигналу у модель системи. Через це інженери змушені вручну підлаштовувати параметри контролера в кожному випадку окремо, аби задовольнити ці обмеження. Дана стаття присвячена розробці альтернативного алгоритму керування на основі підходу керування за прогнозною моделлю. Його основна ідея полягає у тому, щоб генерувати послідовності керування шляхом розв'язання задачі оптимізації, цільова функція якої залежить від передбаченого майбутнього стану. Це дозволяє генерувати швидкі стабілізаційні траєкторії без додаткового підлаштовування алгоритму за рахунок використання еволюційного рівняння класичної лінійної системи як предиктора майбутнього стану, а обмежень на керування — як обмежень задачі оптимізації. Змістовно задана цільова функція критично необхідна, аби цей алгоритм працював належним

чином. Як виявилося, задати цільову функцію з достатньо добрими властивостями є нетривіальною задачею. У даній статті застосовано сучасний нестандартний аналіз, аби зробити це можливим.

Ключові слова: лінійна система, оптимізація, керування за прогнозною моделлю, стабілізація, нестандартний аналіз, гіпердійсні числа, гіпердійснозначна цільова функція.

1. Gubarev V. F., Mishchenko M. D., Snizhko B. M. Model predictive control for discrete MIMO linear systems. *Advanced control techniques in complex engineering systems: Theory and applications: Dedicated to Professor Vsevolod M. Kuntsevich* / ed. by Kondratenko Y. P., Chikrii A. A., Gubarev V. F., Kacprzyk J. Cham : Springer International Publishing, 2019. P. 63–81. ISBN: 978-3-030-21927-7. DOI: 10.1007/978-3-030-21927-7_4.
2. Mishchenko M. D., Gubarev V. F. Methods of model predictive control for discrete multi-variable systems with input. *Cybernetics and Computer Engineering Journal.* 2020. Mar. Vol. 2020, N 1 (199). P. 39–58. DOI: 10.15407/kvt199.01.039.
3. Mishchenko M. D., Gubarev V. F. Horizon length tuning for model predictive control in linear multi input multi variable systems. *Cybernetics and Computer Engineering.* 2021. Mar. Vol. 2021, N 1 (203). P. 39–59. DOI: 10.15407/kvt203.01.039. URL: http://kvt-journal.org.ua/1634/.
4. Goldblatt R. Lectures on the hyperreals. Graduate Texts in Mathematics. Springer-Verlag New York, Inc, 1998. ISBN: 0-387-98464-X.
5. Henle J. M., Kleinberg E. M. Infinitesimal calculus. Cambridge, Mass. : MIT Press, 1979. ISBN: 9780486151014.
6. Keisler H. J. The hyperreal line. *Real numbers, generalizations of the reals, and theories of continua* / ed. by Ehrlich P. Kluwer Academic Publishers, 1994. P. 207–237.
7. Keisler H. J. Elementary calculus: An infinitesimal approach. Second ed. 2012. Feb. URL: https://people.math.wisc.edu/~keisler/calc.html (online; accessed: 2022-09-10).
8. Andersen M.S., Dahl J., Vandenberghe L. CVXOPT: Python software for convex optimization : 2022. URL: https://cvxopt.org/ (online; accessed: 2022-07-17).

Submitted 02.08.2022