Structurally and Procedurally Simplified Soft Computing for Real Time Control

József K. Tar, Imre J. Rudas, János F. Bitó
Budapest Polytechnic
John von Neumann Faculty of Informatics
H-1081 Budapest, Népszínház utca 8., Hungary
{jktar, rudas, jbito}@zeus.banki.hu

Paul H. Andersson, Seppo J. Torvinen
Institute of Production Engineering
Tampere University of Technology
POBox 589, Tampere, 33101, Finland
{paul.andersson, seppo.torvinen}@pe.tut.fi

Abstract: The present state of development of a new branch of Soft Computing (SC) developed for the adaptive control of a special class of non-linear coupled MIMO systems is reported. Its uniform structures are obtained from certain Lie groups unlike that of the traditional SC approaches. The advantages are: a priori known and very reduced size, increase in lucidity; parameter tuning or learning is replaced by simple and short explicit algebraic procedure. The disadvantage is limited circle of applicability. Convergence considerations are discussed for MIMO and SISO systems, too. Simulation examples are presented for the control of the inverted pendulum by the use of the generalized Lorentzian matrices. It is concluded that the method is promising and probably imposes acceptable convergence requirements in many practical cases.

1. INTRODUCTION

The basic components of Soft Computing were almost completely developed by the sixties. In our days it roughly means a combination or integration of neural networks and fuzzy systems enhanced with high parallelism of operation and supported by several deterministic, stochastic or combined parameter-tuning methods. This tuning is often called "learning".

Its main advantage is that it evades the development of intricate analytical system models. Instead of that typical problem classes has been identified for the solution of which typical uniform architectures has been crystallized (e.g. multilayer perceptron, Kohonen-network, Hopfield-network, etc.). The fuzzy systems, too, usually use membership functions of typical (e.g. trapezoidal, triangular or step-like, etc.) shapes, and the fuzzy relations can also be utilized in a standardized way.

The "first phase" of using these methods, that is identification of the problem class and finding the appropriate structure, normally is relatively easy. The following phase, i.e. determining the necessary size of the structure and fitting its parameters via machine learning is far less easy. For neural networks certain solutions starts from a quite big initial network and apply dynamic pruning for getting rid of the "dead" nodes [1]. An alternative method starts with small network, and the number of nodes is increased step by step (e.g. [2-3]). Due to the possible existence of "local optima", for a pure "backpropagation training" inadequacy of a given number of neurons cannot be concluded simply. To compensate this difficulty "learning methods", also including stochastic elements, were seriously improved in the last decade (e.g. [4-7]).

In spite of this development it can be stated that for strongly coupled non-linear MIMO systems traditional SC still has several drawbacks. The number of the necessary fuzzy rules strongly increases with the degree of freedom and the intricacy of the problem. The same is true for the necessary neurons in a neural network approach. Furthermore, external dynamic interactions on which normally no satisfactory information is available influences the system's behavior in dynamic manner. Both the big size of the necessary structures, the huge number of parameters to be tuned, as well as the "goal" varying in time still mean serious problem.

It seems to be reasonable to risk the supposition that "generality" and "uniformity" of the "traditional SC structures" excludes the application of plausible simplifications which may be characteristic to a whole set of typical tasks. This makes the idea rise that several "simplified" branches of SC could be developed for narrower problem classes if more specific features could be identified and taken into account in the uniform structures.

The first steps in this direction were made in the field of Classical Mechanical Systems (CMSs) [8], while further refinements were published in [9-11], on the basis of principles detailed e.g. in [12]. This approach used the internal symmetry of CMSs, the Symplectic Group (SG) of Symplectic Geometry in the tangent space of the physical states of the system. The "result" of the "situation-dependent system identification" was a symplectic matrix mirroring the effects of the inaccuracy of the rough dynamic model initially used as well as the external dynamic interactions not modeled by the controller.

By considering the problem from a purely mathematical point of view, independently of CMSs, it became clear that all the essential steps used in the control can be realized by other mathematical means than symplectic matrices. SG can be replaced by other Lie
groups defined in a similar manner via some "basic quadratic expression". For proceeding in this line the following step of "generalization" must be done: the convergence properties of the method must be investigated in the case when the Lie group used in the control does not describe any internal physical symmetry of the system to be controlled.

In the sequel these considerations are discussed. Simulation examples are presented for the control of the inverted pendulum by the use of the "Generalized Lorentz Group" (GLG).

2. THE CONTROL PROBLEM IN GENERAL

From purely mathematical point of view the control problem can be formulated as follows: there is given some imperfect model of the system on the basis of which some excitation is calculated for a desired input \( \mathbf{i}^d \) as

\[ e = \varphi(\mathbf{i}^d) \]

The system has its inverse dynamics described by the unknown function \( \mathbf{\Phi} = \varphi^{-1}(\mathbf{y}) = \mathbf{f}(\mathbf{i}^d) \) and resulting in a realized \( \mathbf{\Phi} \) instead of the desired one, \( \mathbf{i}^d \). (In Classical Mechanics these values are the desired and the realized joint accelerations, while the external free forces and the joint velocities serve as the parameters of this temporarily valid and changing function.)

It is evident that normally we can obtain information via observation only on the "net" function \( \mathbf{f}(\cdot) \), and that this function considerably varies in time. Furthermore, we do not have practical tools to "manipulate" the nature of this function directly: we can manipulate or deform its actual input \( \mathbf{i}^d \) in comparison with the desired one in general. The aim is to achieve and maintain the \( \mathbf{i}^d = \mathbf{f}(\mathbf{i}^d) \) state. We can directly manipulate only the nature of the model function \( \varphi(\cdot) \), too.

A more or less similar case often happens in theoretical physics. This case can be called as a kind of "renormalization". For instance, suppose, that it is given a real function \( g() \) for which an iteration is defined as

\[ x_{n+1} = g(x_n) \]

and also is given a hypothetical fixed point pertaining to the so obtained sequence on the basis of some other considerations, \( x^d \). Actually it may happen that the given function \( g() \) does not yield a convergent series, or the series may be convergent but it converges to some other value. For a single-dimensional (Single Input, Single Output or SISO) system there is a possibility for so manipulating \( g() \) with a scalar scaling factor \( \gamma \) as

\[ \varphi(x) = \gamma^{-1} \cdot g(x) \]

that the deformed function yields the desired fixed point. For determining the proper deformation factor the iterative sequence of scaling factors

\[ x'^n = s_{n+1}^{-1} \cdot x'^{n+1} \quad \text{for} \quad s_{n+1} x'^{n+1} = g(x'^n) \]

(1)
can be invented, which may be convergent as \( s_n \to x^d \).

Let us suppose that in the given region \( g() \) is continuous and contractive, that is for arbitrary values \( a \) and \( b \)

\[ \| g(b) - g(a) \| \leq K \| b - a \| \quad 0 < K < 1 \]

In this case for the above sequence it holds that

\[ \| x_{n+1} - x_n \| \leq K \| x_n - x_{n-1} \| \]

(2)

that is a Cauchy sequence can be obtained. For a SISO system this sequence is confined in the set of real numbers, therefore it is also convergent. For non-zero \( x^d \) this means a convergent sequence \( \{ s_n \} \) which trivially yields the proper scaling factor sought for because both sides of the second form of Eq. (1) converges to the same number. It is interesting that the contractive nature of \( g() \) itself guarantees this result.

The above considerations should be extended to SISO systems of finite dimensions in which case the scalar parameters should be replaced by some quadratic and invertible matrices \( \{ S_n \} \).

In this context the first difficulty is that Eq. (1) does not have unique solution for the appropriate matrix, and that the calculation of the inverse of some general invertible matrix is very inefficient from computational point of view.

The second problem is that even if \( g() \) is contractive, from Eq. (2) itself no any conclusion can be obtained for the convergence of the sequence \( \{ S_n \} \). It contains information only for the behavior of the sequence in relation with a special vector \( x^d \).

Both difficulties can be conveniently evaded if special restrictions are imposed on \( \{ S_n \} \):

- let its elements be the members of some special Lie group outlined in the next paragraph; this immediately makes the calculation of the inverse matrices efficient;
- within the mathematical framework of the appropriate Lie groups the ambiguity of the solution still remains; to resolve it simply realizable restrictions can be imposed on the allowed matrices by bringing them as close to the identity operator as possible, from a special point of view.

Though the above considerations seem to be very attractive, from phenomenological point of view they cannot be realized because normally there is no possibility for manipulating the system's response with \( S^{-1} \). Only the input can be deformed, and the output can be measured. This circumstance makes it expedient to
consider possible simple modification of the "renormalization algorithm". For a SISO system let us consider the following sequence:

\[ i_1 = s_1 f(x_0), \ldots, i_{n-1} = s_{n-1} f(x_{n-1}), \text{in which} \]
\[ i_0 = s_0 f(x_0) \text{and } f(x_0) < i_0. \]  

(3)

If e.g. for positive \( f(x) \) is positive, monotone increasing, and there exists a constant \( K \) for which

\[ 1 < \frac{K}{f(x)} < K, \text{and } g(x) = \frac{K}{f(x)} \text{is monotone increasing and } f(x) \text{is not bounded} \]

the following estimation can be done:

\[ i_{n+1} - i_n = \frac{i_0}{f(x_i)} - \frac{i_0}{f(x_n)} \]
\[ = s_n i_n - s_n i_{n+1} = i_0 \left( \frac{i_n}{f(x_i)} - \frac{i_{n+1}}{f(x_n)} \right) > 0 \]

(5)

that is the sequence \( f(i_n) \) is monotone increasing. Consequently the sequence \( s_n = i_0 / f(x_n) \) is monotone decreasing with \( s_n > 1 \). Since \( i_{n+1} = s_n i_n > i_n \), the \( \{s_n\} \) sequence has 1 as lower bound. On the other hand the sequence cannot be stopped till \( i_0 > f(x_i) \), therefore it evidently converges to \( s_n = 1 \).

A trivial example for exemplifying that the given set of functions is not empty is \( f(x) = ax + b \), \( a, b > 0 \), \( \in [0, \infty) \) monotone increasing and bounded.

It is natural that a similar case can be imagined, too, for MIMO systems, in analogy with the modified renormalization algorithm with similar special matrices replacing the scalar multiplication factors as

\[ i_n; S(f(x_n) = i_n; i_0 = S_i; \ldots; S(f(x_n) = i_n; \]
\[ i_{n+1} = s_n i_n; \quad s_n \rightarrow 1 \]  

(6)

In a MIMO case the desired convergence can be guaranteed in several manners. More detailed discussion is presented in Paragraph 4.

3. POTENTIAL LIE GROUPS

In the field of CMSs the above considerations correspond to the following logical steps: 1) use a rough initial dynamic model; 2) calculate the necessary generalized forces for the desired joint coordinate accelerations on the basis of this rough model; 3) observe the realized accelerations; 4) create some convenient algebraic means for mapping the observed behavior to the realized one; 5) use the result of this mapping in the next control step. On the basis of physical considerations for mechanical systems this convenient algebraic means was constructed from the elements of the Symplectic Group.

From a purely mathematical point of view this group is not the only possible mathematical means by which such a task can be conveniently solved. Let \( G \) be a nonsingular quadratic, otherwise arbitrary constant matrix. Let the set \( \{v^{(i)}\} \) a linearly independent full set of vectors corresponding to the dimensions of \( G \). Let this set called "special according to \( G \)" if it satisfies the restrictions

\[ v^{(i)} G v^{(i)} = G_0 \]

(7)

It is trivial that the elements of this set can form the columns of a special matrix \( V \), and that this matrix satisfies the equation

\[ V^T G = G \Rightarrow G^{-1} = V^T G, \]

(8)

that is the calculation of the inverse of such matrices in general is very easy and computationally cost-efficient, and furthermore, these matrices form a group as well as the symplectic matrices. Moreover, such matrices may have the determinant only \( \pm 1 \). If we restrict ourselves to the unimodular sub-group, its generators \( H \) have to satisfy the restriction

\[ GH + H^T G = 0 \]

(9)

by the use of which special Lie-groups can be constructed. The special cases in which \( G \) corresponds to \( I \), \( \mathfrak{g} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \), and \( \begin{pmatrix} 1, 1, 1, -c \end{pmatrix} \), result in the Orthogonal, the Symplectic, and the Lorentz Group, respectively ("c" is the velocity of light). The appropriate special sets are the orthonormal, the symplectic, and the Lorentzian sets. In these examples \( G \) is either symmetric (\( I, g \)) or skew-symmetric (3), consequently \( H \) can be constructed of skew-symmetric or symmetric \( J \) matrices, respectively, as \( H = G^{-1} J \).

All the considerations used for constructing a mapping between the observed and the desired behavior can trivially be repeated in the case of another groups, supposing, that at least one element of the special sets can be an arbitrary non-zero vector. This definitely does not holds for the Orthogonal Group because the appropriate sets consist of pairwise orthogonal unit vectors. Therefore, instead of the orthogonal group the combined group of positive scalings and rotations can be used which we call the "Stretched Orthogonal Group".

For using the "Generalized Lorentz Group" exempt of this difficulty, first a "fictitious dimension" can be "added" to the DOF dimensional problem. So for \( G \) the diagonal matrix \( \mathfrak{g} = \begin{pmatrix} 1, \ldots, 1, -c^2 \end{pmatrix} \) can stand. Now let the DOF dimensional vector \( f \) stand for the desired/observed
joint coordinate acceleration, and let us start with the columns of the DOF×DOF dimensional unit matrix. In the first step let this set be rigidly so rotated that its first vector becomes parallel with \( \mathbf{f} \). It is easy to so construct the rotation operators that the orthogonal sub-space of the initial and the goal vectors remains unchanged. Let \( \mathbf{e}^{(j)} = \mathbf{f} / \sqrt{\mathbf{f} \cdot \mathbf{f}} = \mathbf{f} / \mathbf{f} \). It is trivial that the columns of the following matrix form a generalized Lorentzian set:

\[
\begin{bmatrix}
\mathbf{e}^{(1)} \sqrt{f^2 / l^2 + 1} & \mathbf{e}^{(2)} & \cdots & \mathbf{e}^{(DOF)} & \mathbf{f} \\
\mathbf{f} / l^2 & 0 & \cdots & 0 & \sqrt{f^2 / l^2 + 1}
\end{bmatrix}
\] (10)

In this solution the physically interpreted vector \( \mathbf{f} \) is accomplished with a fictitious (DOF+1)th component, and it is placed into the last column of a generalized Lorentzian. In strict analogy with Eq. (8) the proper Lorentzian transforming the observed acceleration into the desired one can be calculated as

\[
\mathbf{L} = \begin{bmatrix}
\mathbf{f}^{(1)} / \sqrt{f_{D^2} / l^2 + 1} & \cdots & \mathbf{f}^{(DOF)} / \sqrt{f_{D^2} / l^2 + 1} & \mathbf{g}
\end{bmatrix} \mathbf{g}^{-1}
\] (11)

For control-technical purposes "c" may be an arbitrary positive constant. Since several results already were published for the Symplectic Group and the Stretched/Shrunken Orthogonal Group, the simulations in the present paper are concentrated of the use of the Generalized Lorentz Group.

4. STABILITY CONSIDERATIONS

The \( \mathbf{f}(\mathbf{i}) \rightarrow \mathbf{i}_p \) requirement can be expressed in more or less restricted forms. For instance, assume, that there exists \( 0 < K < 1 \) for which

\[
\| \mathbf{f}(\mathbf{i}) - \mathbf{i}_p \| < K \| \mathbf{f}(\mathbf{i}) - \mathbf{i}_p \| < \cdots < K^L \| \mathbf{f}(\mathbf{i}) - \mathbf{i}_p \| \leq \| \mathbf{f}(\mathbf{i}) - \mathbf{i}_p \| (13)
\]

This requirement trivially guarantees the desired convergence. The question is whether it can be satisfactory (that is not too rigorous) from practical point of view. Let the \( \mathbf{S} \) matrices in Eq. (6) be written in the form of \( \mathbf{S} =\mathbf{I} + \mathbf{C} \). The desired convergence now means that \( \mathbf{C}_{\mathbf{i} \rightarrow \mathbf{i}_p} \rightarrow 0 \). Consider a differentiable function \( \mathbf{f}(\mathbf{x}) \), and three points in the space: \( \mathbf{f}(\mathbf{x}), \mathbf{f}(\mathbf{Sx}), \) and \( \mathbf{Sf}(\mathbf{x}) = \mathbf{i}_p \). In the iteration the present error, and the next error can be expressed as

\[
\mathbf{H} = \mathbf{f}(\mathbf{x}) - \mathbf{i}_p = \mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{x}) + \mathbf{f}^{(x)}(\mathbf{f}(\mathbf{x})) = -\sigma(\mathbf{f}(\mathbf{x}))
\]

\[
\mathbf{H} = \mathbf{f}(\mathbf{x} + \Delta \mathbf{x}) - \mathbf{i}_p = \mathbf{f}(\mathbf{x}) + \mathbf{D}\sigma - \mathbf{i}_p = \mathbf{D}\sigma - \sigma(\mathbf{f}(\mathbf{x})) (14)
\]

In Eq. (14) \( \mathbf{D} \) corresponds to the derivative of \( \mathbf{f}(\mathbf{x}) \) for near zero \( \sigma \) matrices. For these

\[
\frac{\| \mathbf{H}^2 \|}{\| \mathbf{H} \|} = \frac{\mathbf{x}^{T} \sigma^{T} \mathbf{D} \left( \mathbf{D}\sigma - 2\sigma(\mathbf{f}(\mathbf{x})) \right) + \mathbf{f}(\mathbf{x})^{T} \sigma^{T} \mathbf{f}(\mathbf{x})}{\mathbf{f}(\mathbf{x})^{T} \sigma^{T} \sigma(\mathbf{f}(\mathbf{x}))} \leq K^2 < 1
\]

is needed. Now suppose that \( \mathbf{f}(\mathbf{x}) \) is in the same order of magnitude as \( \mathbf{x} \), and that the function is "very flat" in the sense that its derivative \( \mathbf{f}(\mathbf{x}) \) is very small. In this case the first term in the numerator of Eq. (15) can be negligible in comparison with the second one and the additional requirement \( \mathbf{x}^{T} \sigma^{T} \mathbf{D} \sigma(\mathbf{f}(\mathbf{x})) > 0 \) may lead to the desired convergence.

This does not seem to be too unrealistic in many physical cases. Consider e.g. the classical mechanical systems in which \( \mathbf{x} \) corresponds to the joint acceleration and \( \mathbf{f}(\mathbf{x}) = \mathbf{M}^{-1} \mathbf{Mx} + \mathbf{M}^{-1} \left( \mathbf{b} - \mathbf{b} \right) = \mathbf{Ax} + \mathbf{c} \) in which \( \mathbf{M} \) and \( \mathbf{b} \) correspond to the model inertia and the Coriolis plus gravitational terms, respectively, while their counterparts denoted by the tilde symbol correspond to the real data. With small \( \mathbf{M} \), that is with under-estimated model inertia, and a model term \( \mathbf{b} \) which can be arbitrarily set, the fulfillment of this requirement is not hopeless. In the next paragraph simulation examples are presented for the most "popular" paradigm, the control of the inverted pendulum.

5. SIMULATION RESULTS

The inverted pendulum has the usual structure with one linear and one rotational degree of freedom: it consists of a carrier of mass of 50 kg being able to move in linear direction, a rotatable rod of 1 m length and negligible mass, and a point-like mass on its end of 10 kg. The gravitational acceleration was supposed to be 10 \( \text{m/s}^2 \). The parameter \( c = 1 \) was chosen in the Lorentzians. Instead of varying it, a weighting factor \( w = 1000 \) was chosen for "scaling" the joint accelerations (via division) before putting them into the Lorentzians. The proper part of the resulting Lorentzian was multiplied by this factor before using it in the control. The proper part of the control programme was realized as follows:

\[
U = \text{Lorn} \ast \text{lor1}(qDppMod/\text{weight},c);
\]

\[
Q = \text{Modln1}*\text{weight}*U(1:2,3) + \text{Bmod};
\]

In the Lorentzian \( \text{Lorn} \) the result of the previous identification is stored, while function \( \text{lor1} \) creates the Lorentzian from the weighted input according to Eq. (11). In the second row the defomed input is obtained from the proper blocks of the resulting matrix, it is scaled to the physically desired range by the weight factor, an is put into the rough dynamic model (represented by the model
inertia matrix $M(\text{ini1})$) to obtain the required generalized forces $Q$.

Three kinds of control are compared: a kinematically prescribed PD control with the \textit{exact dynamic model}, the same kinematically prescribed PD control with the rough dynamic model \textit{without any adaptation}, and the same kinematic requirements with adaptive control realized by the generalized Lorentzian matrices.

In Fig. 1 typical results are given regarding trajectory reproduction. It is evident that the rough initial model is quite wrong and that its adaptive accomplishment well approaches the CTC control based on the exact dynamic model.

To reveal technical details in Fig. 2 the norm of the appropriate generalized Lorentzians and the joint acceleration errors are displayed. In the adaptive case these norms are not constant but they are very close to the norm of the identity transformation of the $3\times3$ matrix ($\equiv 1.732$) revealing that in the iteration the $S$ matrices were really close to the identity transformation. (In the case of the exact CTC control the Lorentzians were the unit matrices and the acceleration error was in the order of magnitude of $10^{-13}$, that is within the precision of the computations.)

In Fig. 3 the graphs of the generalized forces are given for the exact model-based CTC control and the adaptive control in correlation with the trajectory reproduction error. In the adaptive case (switched on in the 50th time unit) the generalized forces have smooth variation in time too.
6. CONCLUSIONS

In this paper the possibility of developing a particular special branch of Soft Computing was investigated in which the uniform structures to be used originate from different abstract Lie-groups. The new approach has two essential advantages in comparison with the "traditional means" of Soft Computing:

- The structure-size and the number of the free parameters is uniquely determined by the degree of freedom of the system to be controlled and the particular group chosen;
- Machine learning can be realized via simple, deterministic, definite algebraic steps limited in number; It is void of the problem of local optima.

The convergence properties of the learning algorithm were investigated in general. A possible particular criterion was suggested which seems to be met in practical cases as classical mechanical systems' control.

The applicability of the approach was demonstrated via simulation in the case of the inverted pendulum, controlled by the use of the Generalized Lorentz Group.

It can be expected that the here presented considerations can be extended to a wider class than the control of mechanical systems. In general seeking for different convergence criteria for the learning seems to be expedient.

7. ACKNOWLEDGMENT

The authors gratefully acknowledge the aid by the Finnish-Hungarian Bilateral Technology Co-operation Project (2000-2001) supporting their work.

8. REFERENCES


