



HAL
open science

Complex Systems Modelling Developed for Electromagnetic Compatibility Applied to Automatic Problems

Olivier Maurice, Philippe Durand

► **To cite this version:**

Olivier Maurice, Philippe Durand. Complex Systems Modelling Developed for Electromagnetic Compatibility Applied to Automatic Problems. 2016. hal-01345514

HAL Id: hal-01345514

<https://hal.science/hal-01345514>

Preprint submitted on 13 Jul 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Complex Systems Modelling Developed for Electromagnetic Compatibility Applied to Automatic Problems

Olivier Maurice^{1,*}, Philippe Durand²,

1 univ. Rouen, UNIROUEN, ESIGELEC, IRSEEM 76000 Rouen, France

2 CNAM, M2N, 75000 Paris, France

* olivier.maurice@esigelec.fr

Abstract

Gabriel Kron has developed tensorial analysis of networks (TAN) in 1939. This formalism gives all the techniques to analyse systems. Extending his work, one has submitted the xTAN method for "extended tensorial analysis of networks" adding chords and game theory [1], firstly focused on electromagnetic compatibility, the authors not being in the automatic community. But it seems that the method developed may be interesting for this community, giving perhaps new opportunities in system description as in their modelling.

Mathematical considerations and illustrations on examples are given in order to present as clearly as possible the formalism and its application to automatic. Future works, under the assumption that the method presents interesting sides, will consist in applying the method to various case in systems modelling, in order to improve its performances.

Introduction

Gabriel Kron has developed the tensorial analysis of networks. Rather than using nodes and edges in graphs to describe the circuit topology, he developed circuit definition using the space of the meshes. This approach was justified through topology considerations. The fact is that Kron's method leads to rigorous description of complex systems, for example electrical machines: Kron's model for an electrical machine is well known to be the more general and accurate one. At this time, there had no need for a generalized interaction technique. In automatics, the classical technique to represent the system uses diagrams. They are similar to graphs, with different drawing conventions. The fundamental added value of Kron's approach is to use topological considerations, including dual space and metric tensor. To study complex systems, Kron's method lacks of generalized interactions that we submit and named "cords" [2]. We apply these techniques here for the automatics and we make a coupling between the method of Kron and game theory, thinking that it perhaps gives some new opportunities for complex systems modelling. This coupling to game theory was historically first inspired by mathematical tools for cybernetics [4] and economical works [5]. The objective is to give theoretical modelling capabilities to include the human factor. We present here some of the tools developed by the author in order to submit a complete technique covering both an automatic aspect and a human factor aspect in systems modelling.

Notations

Throughout the paper, we use Einstein's mute index notation. It means that each time an index is repeated, it implies a summation operation on the values of this index, i.e.:

$$\begin{cases} \sum_{\alpha} a_{\alpha} u^{\alpha} \equiv a_{\alpha} u^{\alpha} \\ \sum_{\beta} \Gamma_{\nu\beta} x_{\beta} \equiv \Gamma_{\nu\beta} x_{\beta} \end{cases} \quad (1)$$

Vectors projected in a base \vec{b}_k are identified by their components: $\vec{u} = u^k \vec{b}_k \rightarrow u^k$, with the index up, and covectors $\alpha(\vec{u} \cdot \vec{b}_k) = \omega_k$ by their components with index down.

First mathematical considerations

General variables $x^k \in \mathbb{R}^n$, $k \in \mathbb{N}$ involved in any system are seen as intensities of displacement in an n -dimensional space \mathbb{V} of base \mathbf{b}_k . One vector identifies these displacements at any time $\mathbf{v} = x^k \mathbf{b}_k$. \mathbf{v} takes in charge all the flux inside the system (including information, mechanical, electrical flux, etc.).

Principles

Whatever the flux, it means that some solicitation (effort) exists having created this flux. We can define a dual space \mathbb{V}^* , by taking in charge all these solicitations of cobase \mathbf{e}^k . By definition $\mathbf{e}^k \mathbf{b}_m = \delta_m^k$, δ_m^k being Kronecker symbol. Any energy or information source, etc., is defined by a covector in this cobase $v^* = u_k \mathbf{e}^k$. To go from one vector to one covector, the only solution goes through the fundamental tensor $z \in \mathbb{R}^n \times \mathbb{R}^n$, which is defined by the invariant of power S (S is a scalar):

$$u_k = z_{km} x^m \Rightarrow x^k u_k = x^k z_{km} x^m = S \quad (2)$$

Note that:

$$v^* \mathbf{v} = u_n \mathbf{e}^n x^n \mathbf{b}_n = u_n x^n \delta_n^n = S \quad (3)$$

We see that working directly with the components leads to the same result as if we made all development start from vectors and covectors.

The tensor z is the fundamental tensor of the theory. For each problem, we choose a tensor that describes physical relations from the graph that represents the problem. The graph is an intermediate action that helps establish the equations. The most simplest tensor that can be taken is unity. In this case $u_k = x^k$. But more complex ones can be used, in particular the tensors where components are operators. In this case $u_k = z_{km}(x^m)$. When the relation between the flux u_k and the effort x^k is 1, it means that we can make the drawing of a simple network for which the source of flux is a generator and the flux is a current. The relation 1 is an intrinsic function attached to the mesh saying that $u_1 = 1.x^1$, for the first mesh of flux x^1 and effort u_1 (see graph figure 1). The fact to

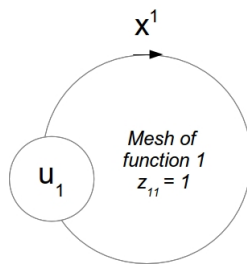


Figure 1: Elementary mesh as graph

consider an elementary structure like a simple mesh as able to wear a flux comes from the fact that primitive components can be defined at the edge level [3]. For the needs of the automatics, we can work directly in the meshes space, without looking at the edges space for the moment. But having other space levels (edges, nodes, node pairs, moment) may be an advantage for future problems where descriptions of higher complexity in graphs are necessary.

z structure

Once fundamental relations are chosen between flux and efforts, control loops are taken into account through cords. A cord is a function that links one flux $x^{(i)}$ and one effort $u_{(j)}$ (the parenthesis indicates that the index points out a particular component). This effort is created by the flux through the extra-diagonal component of the metric $z_{(ji)}$:

$$u_{(j)} = z_{(ji)}x^{(i)} \quad (4)$$

If a control loop returns part of the information from an output to some input, it should be taken in charge by a cord adding an effort to some circuit, attached with some output flux. A basic output $y \in \mathbb{R}$ given by some dependence with an input $x \in \mathbb{R}$, a gain $G \in \mathbb{R}$ and a control loop $f(\bullet) : \mathbb{R} \rightarrow \mathbb{R}$, written $y = G(x + f(y))$ can be replaced by the network shown in figure 2 using two meshes. The cord of function G sends G times the flux of the first mesh to the second mesh. The flux of the first mesh is its total effort divided by its impedance function, here unitary:

$$x^1 = \frac{u_1 + f(x^2)}{z_{11}} = \frac{u_1 + f(x^2)}{1} = u_1 + f(x^2) \quad (5)$$

Now the effort reported on the second mesh is

$$u_2 = z_{21}x^1 = Gx^1 = G(u_1 + f(x^2)) \quad (6)$$

as $y = z_{22}x^2 = u_2$ (because $g_{22} = 1$) and $u_1 = x$ we obtain the expected result.

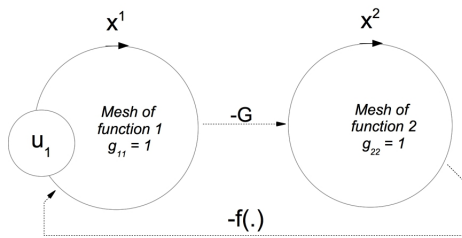


Figure 2: Network of elementary controlled loop

Extra-diagonal components of the metric are G for z_{21} and $f(\bullet)$ for z_{12} . Finally, its matrix is:

$$z = \begin{bmatrix} 1 & -f(\bullet) \\ -G & 1 \end{bmatrix} \quad (7)$$

and the effort covector \mathbf{v}^* is given by:

$$\mathbf{v}^* = \begin{bmatrix} u_1 \\ 0 \end{bmatrix} \quad (8)$$

The system is completely defined by the equation $\mathbf{v}^* = z \cdot \mathbf{v}$ or

$$u_k = z_{km} \cdot x^m \quad (9)$$

The dot symbol is here to recall that g can have operators, like the function f . It's not a simple product but the application of z to the flux \mathbf{v} . Developing, it gives:

$$\begin{cases} u_1 = x^1 - f(x^2) \\ 0 = -Gx^1 + x^2 \end{cases} \quad (10)$$

Which is equivalent to previous relations ($u_2 = Gx^1 = G(u_1 + f(x^2))$).

Once the graph is drawn, it's very easy to conclude on z structure. Each mesh k generates 1 as z_{kk} component, and cords define all the extra-diagonal components when they exist. If not, $z_{km} = 0$. For example the graph given in figure 3 is associated with the metric:

$$z = \begin{bmatrix} 1 & -L & -M & 0 \\ G & 1 & 0 & 0 \\ 0 & h & 1 & -B \\ a & 0 & k & 1 \end{bmatrix} \quad (11)$$

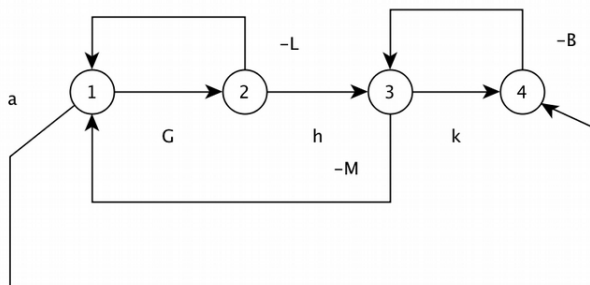


Figure 3: Example of graph

Now if the output is $y = x^4$ and the single input is $u_1 = z_{11}x^1$, the expression of y is immediate:

$$y = z^{-1} \begin{bmatrix} u_1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (12)$$

Note that the equation $u_a = z_{ab}x^b$ is directly equivalent to Lagrange's equations of the problem [6]. The system of equations is an integro-differential system and, more than that, it can include non linear operators.

Illustration with a canonical case in automatic

We consider a motor system made of one "RL" circuit and an electrical motor. The modelling is the classical representation presented in automatic courses. If C_m is the motor couple and C_r the resistive couple, K_d the constant of friction, K_e the electromotive constant, K_t the electromagnetic couple and J the inertia moment, we write:

$$\begin{cases} C_m = K_t i \\ K_t i - K_d w - C_r = J \frac{dw}{dt} \end{cases} \quad (13)$$

w is the angular speed.

Knowing these equations and seeing the next diagram (figure 4) we obtain the following equation system:

$$\begin{cases} u = (R + Lp) i - K_e w \\ -C_r = -K_t i + (K_d + Jp) w \end{cases} \quad (14)$$

with p Laplace's operator.

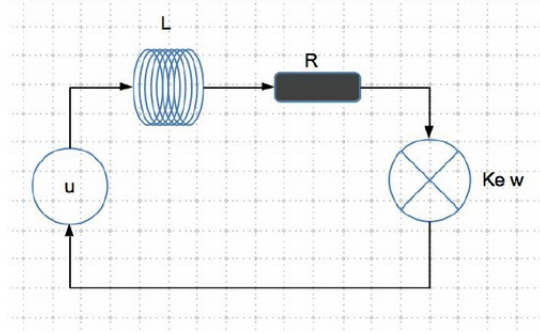


Figure 4: Motor control diagram

With some use, the engineer may have drawn the diagram figure 5. From our previous discussion,

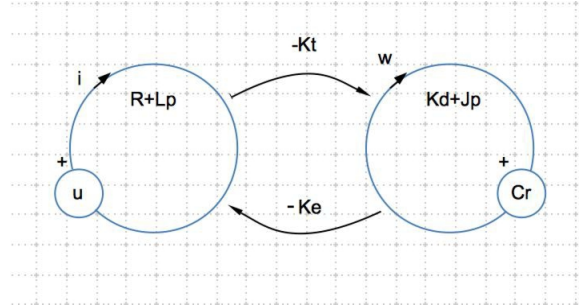


Figure 5: xTAN technique diagram for the motor control

we deduce from this graph the following fundamental tensor:

$$z = \begin{bmatrix} R + Lp & -K_e \\ -K_t & K_d + Jp \end{bmatrix} \quad (15)$$

associated with the covector $[u \quad -C_r]$.

Theoretical approach: second mathematical considerations

One can wonder what are the advantages of this formalism compared to classical formalisms? At this level of development, it is impossible to compare the various methods. But the formalism submitted here leads to geometrical considerations. The system is represented into a cellular topology represented through a graph. In this graph, meshes are identified as major elements to give similar relations to the bloc diagrams for the automatics. Working in the meshes space gives special advantages [7]. We first take a look at the manifold concept then we give the key connecting the system description to a differential geometry.

Sub-manifolds

The definitions below are well known, but we want to fix ratings. A variety X of dimension n is a topological space that locally looks like an Euclidian space. We mean by that every neighborhood of X can be sent using a homeomorphism on a neighborhood of \mathbb{R}^n . φ is also called a chart.

This notion led to the definition of abstract topological or smooth manifolds by chart, differentiable atlas Historically it is not by this meaning that manifolds have been defined. The manifolds (sub manifolds) are introduced as (topological) subsets of an Euclidean space. Moreover, it is in the category of differentiable objects, that most applications can be defined. By the way, it is by trying to adapt the differential calculus by nature local, at global objects it could be defined the notion of smooth sub-manifold of \mathbb{R}^n . It is this point of view that is adopted here and not the background of the intrinsic geometry. This is also close to the definition given by Poincaré [8].

We say that a topological space M is a sub-manifold of \mathbb{R}^n if a neighborhood of M can be sent smoothly to a neighborhood of \mathbb{R}^m . More precisely, M is a submanifold of dimension m , if there exists a neighborhood U of x in M and a neighborhood V of 0 in \mathbb{R}^n , a diffeomorphism φ sending U on V with: $\varphi(U \cap M) = V \cap (\mathbb{R}^m \times \{0\})$ There are basically two methods and to describe sub manifolds. By equations (also called constraints in the language of the engineer that is adopted here) or by parameterization. it is connected to the concepts of immersion and submersion, The interested reader can go back to the classic reference textbooks of differential geometry. Thus simple example is given by the circle, and can be described from $t \rightarrow (cost, sint)$ (parameterization) or by equation $(x^2 + y^2 - 1 = 0)$. In the following, the differentiable manifolds, will be understood as sub-manifolds of \mathbb{R}^n defined by equations.

To join our concept of sub-manifold in the previous example, we should detail the interval to which the variables belong. For example $u \in [u_{min}, u_{max}]$. If we note in general the covector u_a , for a set of variables x^b , the system equation associated with some graph is:

$$\begin{cases} u_a = z_{ab} (x^b) \\ x^1 \in [x_{min}^1, x_{max}^1] \\ \dots \\ x^{(n)} \in [x_{min}^1, x_{max}^1] \end{cases} \quad (16)$$

$u_a, x^b \in \mathbb{R}^n, z \in \mathbb{R}^n \times \mathbb{R}^n$. This system of equations is the definition of a sub-manifold \mathcal{M}_i , inspired by the original one given by Poincaré [8]. The intervals can be noted $I(x^{(k)})$ for the component $x^{(k)}$ and the same for all variables. The previous sub-manifold definition becomes:

$$\mathcal{M}_i = \begin{cases} u_a = z_{ab} (x^b) \\ I(u_1), \dots, I(x^{(n)}) \end{cases} \quad (17)$$

Knowing such a system, a more complex one can be built using others also known under the same definitions, making a global manifold \mathcal{M} . The fundamental tensor g of the constructed system results from direct summation of each fundamental tensor $z_i, i \in \{1, 2, \dots, N\}$ associated with each sub-manifold \mathcal{M}_i . i.e.:

$$z = \bigoplus_{k=1}^N z_k \quad (18)$$

For intervals, the summation consists in adding intervals of each sub-manifolds in the global one:

$$I_{\mathcal{M}} = \{I_{\mathcal{M}_1}, I_{\mathcal{M}_2}, \dots, I_{\mathcal{M}_N}\} \quad (19)$$

If we consider two sub-manifolds \mathcal{M}_1 and \mathcal{M}_2 of variables $\{x^1, x^2\} \in \mathcal{M}_1$ and $\{x^3, x^4\} \in \mathcal{M}_2$; making a new system based on these two elementary ones can be realized in two steps:

- to make the direct summation of the various sub-manifolds involved;
- to add coupling - control loops that link all the sub-manifolds involved.

For example we consider two systems of sub-manifolds \mathcal{M}_1 and \mathcal{M}_2 presented in figure 6 (graph realized with Yed software). The fundamental tensors of these two networks are easily found:

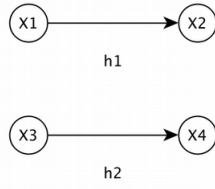


Figure 6: Two separated networks

$$z_1 = \begin{bmatrix} 1 & 0 \\ h_1 & 1 \end{bmatrix} \quad z_2 = \begin{bmatrix} 1 & 0 \\ h_2 & 1 \end{bmatrix} \quad (20)$$

Associated interval definitions fix the variable values belonging: $x^1 \in I(x^1), \dots, x^4 \in I(x^4)$. The first step is to make the direct summation of z_1 and z_2 :

$$z = z_1 \oplus z_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ h_1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & h_2 & 1 \end{bmatrix} \quad (21)$$

To connect these two networks means to add control loops - i.e. cords - between at least two g components. For example a link between variables x^3 and x^2 could be added as shown in figure 7.

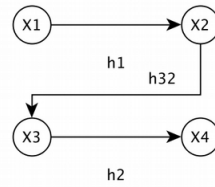


Figure 7: Added connection between 3 and 2

The fundamental tensor becomes:

$$z = z_1 \oplus z_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ h_1 & 1 & 0 & 0 \\ 0 & h_{32} & 1 & 0 \\ 0 & 0 & h_2 & 1 \end{bmatrix} \quad (22)$$

Many other possibilities (10 here) can be used to increase the interactions between preliminary sub-manifolds.

Second "geometrization" process

Applying a transformation Λ to the fundamental tensor of the rotation group, we can hope to obtain a unique diagonal expression of it:

$$\Lambda_\alpha^a z_{ab} \Lambda_\beta^b = \zeta_{\alpha\beta}, \text{ with } \zeta_{\alpha\beta} = 0, \text{ if } \alpha \neq \beta \quad (23)$$

This mechanism is possible if and only if z is symmetric. We have seen that usually, coming from automatic diagrams, it results that g is not symmetric.

One has submitted a solution to transform z in order to make it symmetric [9] [10]. We consider the system $u_\alpha = z_{\alpha\beta} x^\beta$ as a parametrized N-dimensional surface. Each source $u_\alpha(x^1, \dots, x^{(n)})$

being equal to a function $\psi_\alpha(x^1, \dots, x^{(n)})$. Calculating partial derivatives of $\psi = [\psi_\alpha]$ relative to x^β creates a base of vectors:

$$\mathbf{b}_k = \frac{\partial \psi}{\partial x^k} \quad (24)$$

Using these vectors like columns of a new matrix, it leads to the jacobian $\mathcal{J}_{mk} = [\mathbf{b}_k]_m$. Making the comparison between both equations $u_\alpha = z_{\alpha\beta}x^\beta$ and $u_\alpha = \mathcal{J}_{\alpha\beta}x^\beta$ we see that it lacks components. Each time a time derivative is included as an operator in g , it leads to zero:

$$\frac{\partial}{\partial x^k} \left(L_{mk} \frac{dx^k}{dt} \right) = L_{mk} \frac{d}{dt} \frac{\partial}{\partial x^k} x^k = 0 \quad (25)$$

with $g_{mk}(\bullet) = L_{mk}d/dt(\bullet)$. So we can write:

$$g_{\alpha\beta}x^\beta = (\mathcal{J}_{\alpha\beta} + \mathcal{L}_{\alpha\beta})x^\beta \quad (26)$$

and, as a consequence, our system equation can now be written:

$$u_\alpha - \mathcal{L}_{\alpha\beta} \frac{d}{dt} x^\beta = \mathcal{J}_{\alpha\beta} x^\beta \quad (27)$$

We can multiply both members by $\Gamma_{\zeta\alpha}$ which is the transposed matrix of $\mathcal{J}_{\alpha\beta}$: $\Gamma_{\zeta\alpha} = \mathcal{J}_{\alpha\beta}^T$. But there is the remarkable relation:

$$\Gamma_{\zeta\alpha} \mathcal{J}_{\alpha\beta} = G_{\zeta\beta} \quad (28)$$

G is called a metric and is defined by $G_{ij} = \langle \mathbf{b}_i, \mathbf{b}_j \rangle$.

Noting $T_\zeta = \Gamma_{\zeta\alpha} (u_\alpha - \mathcal{L}_{\alpha\beta} \frac{d}{dt} x^\beta)$ the source-inertia tensor, the system equation becomes:

$$T_\zeta = G_{\zeta\beta} x^\beta \quad (29)$$

G is a symmetric tensor and eigenvalues can now be find for the problem. This is one advantage of this new formulation submitted by the author. But there is another one: the formalism is now completely included in differential geometry rules [11]. For example, derivation of basic vectors can be calculated:

$$\mathbf{b}_{km} = \frac{\partial}{\partial x^k} \mathbf{b}_m \quad (30)$$

If the fundamental tensor z component depends itself on variables x^k , this derivative is different from zero. The basic vectors \mathbf{b}_k are the vectors of a tangential plan $T_p S$ to the surface ψ and change from one location to another on this surface, as its vectors depend on the flux values x^β . The projection of these vectors \mathbf{b}_{mk} on the plan axes leads to the scalar product:

$$\Gamma_{km,q} = \langle \mathbf{b}_{km}, \mathbf{b}_q \rangle \quad (31)$$

$\Gamma_{km,q}$ are Christoffel's coefficients. They appear on specific theoretical studies. For example if we analyse dependencies of T versus x^k , as:

$$\frac{\partial}{\partial x^k} \mathcal{L}_{\alpha\beta} \frac{d}{dt} x^\beta = \mathcal{L}_{\alpha\beta} \frac{d}{dt} \frac{\partial}{\partial x^k} x^\beta = 0 \quad (32)$$

assuming $\Gamma_{\zeta\alpha}$ and $\mathcal{L}_{\alpha\beta}$ is constant, we obtain:

$$\frac{\partial}{\partial x^k} T_{\zeta\beta} \equiv \frac{\partial}{\partial x^k} u_\zeta = \frac{\partial}{\partial x^k} G_{\zeta\beta} x^\beta \quad (33)$$

but:

$$\frac{\partial}{\partial x^k} G_{\zeta\beta} = \frac{\partial}{\partial x^k} \langle \mathbf{b}_\zeta, \mathbf{b}_\beta \rangle = \Gamma_{k\zeta,\beta} + \Gamma_{k\beta,\zeta} \quad (34)$$

The command dependencies can be theoretically studied though Christoffel's coefficients. When they are null for example, it means that the command does not depend on x^k values.

Sub-space to mesh one

The x^k can be seen as curvilinear coordinates and u_k as the local projection on the T_pS tangent plan.

Dimensions of G depend on the flux involved. In electromagnetism, G is in volts. Its study is very interesting in order to have a geometrical interpretation of the system behaviour. But the submitted formalism does not imply the space configuration choice. If we want to introduce variables that correspond to flux integration, they are attached to nodes. In other words they belong to N cellular topological space and $\chi^k = \dot{q}^k$, where χ is edge's flux. But we can also say that to one node, the summation of all the fluxes and the static loads that belong to the node is zero:

$$q^k = A_m^k \chi^m, q \in N, \chi \in E \quad (35)$$

A is classically called the incidence matrix. Through A we go from the nodes (space N) to the edge: space E . Now if we create a closed circulation of edges, we can make a mesh. This means that through a connectivity B we can go from the edge space E to the mesh space M using:

$$\chi^k = B_m^k x^m, \chi \in E, x \in M \quad (36)$$

Kirchhoff's edge can be generalized and groups all these quantities. Figure 9 shows Kirchhoff's edge.

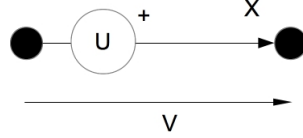


Figure 8: Kirchhoff's edge

For a system where n edges are used separately, we can write:

$$U_a = Z_{aa} \chi^a + V_a \quad (37)$$

The potential energy is seen through the potential difference V while the command or the energy source U give the flux χ . This flux, added to the command U creates the potential difference V for a given function Z . This potential difference V is linked with the mass q through $q^k = \gamma^{km} V_m$. When using meshes, we use the connections defined in B . From the previous relation we can write:

$$U_a = Z_{aa} B_\alpha^a x^\alpha + V_a \quad (38)$$

Multiplying each member by the transposed connectivity B' we obtain:

$$B_\sigma'^a U_a = B_\sigma'^a Z_{aa} B_\alpha^a x^\alpha + B_\sigma'^a V_a \quad (39)$$

but due to the fact that the potential energy has a null work on a closed circulation, $B_\sigma'^a V_a = 0$. As $B_\sigma'^a Z_{aa} B_\alpha^a = z_{\sigma\alpha}$ and noting $u_\sigma = B_\sigma'^a U_a$, we finally obtain:

$$u_\sigma = z_{\sigma\alpha} x^\alpha \quad (40)$$

Which is the well known Kron's equation in the mesh space [12]. But we can add self properties s of meshes to complete the z tensor - for example adding self inductance to a loop:

$$z_{\sigma\alpha} \rightarrow z_{\sigma\alpha} + s_{\sigma\alpha} |_{\sigma=\alpha} \quad (41)$$

Next step is to add interactions between meshes or edges of meshes Q :

$$z_{\sigma\alpha} \rightarrow z_{\sigma\alpha} + Q_{\sigma\alpha} |_{\sigma \neq \alpha} \quad (42)$$

This last fundamental tensor z is the one used in equation (9), having made its proofs in various applications for the electromagnetic compatibility [13].

The various topological descriptions of space relations was accurately detailed by Roth [14] [15] [16]. Critics were formulated on the tensorial aspect of Kron's object. This was discussed by Hoffmann [17] and recently by the author [18] [19]. In case of non linear operators, it seems clear that Kron's formalism as the second geometrization process lead to objects that does not belong to the classical definition of riemannian spaces. In all cases, tensorial equations are in direct relation with Lagrange's equations, as shown below for Kron's formulation.

From Lagrange's equations to Kron's ones

Under our hypothesis and using Laplace's operator p , laws associated with edges become: $E_0 = Ri$, $E_0 = Lpi$, $E_0 = 1/(Cp)i$. If M, B, N, R are the topological characteristic of one circuit, we represent by $i^a, a = 1, 2, \dots, M$ the mesh currents associated with M meshes chosen in the circuit. $e_k, k = 1, 2, \dots, M$ are the meshes electromotive forces. With variables i^k we can associate mesh loads q^k with $i^k = \dot{q}^k$. Figure 9 shows two coupled meshes where inductances, elastances, resistances are associated with branches or meshes, mesh currents and loads and an example of cord for mutual inductance between meshes. In the whole circuit, cinetic energy is given by:

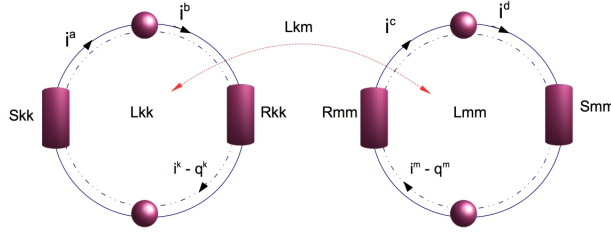


Figure 9: Example of network for lagrangian

$$T = \frac{1}{2} \mathcal{L}_{km} \dot{q}^k \dot{q}^m \quad (43)$$

For the potential energy we have:

$$\mathcal{U} = \frac{1}{2} S_{km} q^k q^m \quad (44)$$

where S_{km} are elastance values (inverse of capacitance). And finally for dissipations:

$$\mathcal{W} = \frac{1}{2} R_{km} \dot{q}^k \dot{q}^m \quad (45)$$

Lagrange's equations for the circuit are given by:

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}^k} \right) + \frac{\partial \mathcal{U}}{\partial q^k} + \frac{\partial \mathcal{W}}{\partial \dot{q}^k} = e_k \quad (46)$$

With (3),(4),(5) and using Laplace's operator p and transformation on i and e , with i_0^k initial conditions in current and q_0^k initial conditions in loads, Lagrange's equation becomes:

$$\mathcal{L}_{km} p (i^m - i_0^m) + \mathcal{R}_{km} i^m + \mathcal{S}_{km} \left(\frac{i^m}{p} + q_0^m \right) = e_k \quad (47)$$

Defining: $Z_{km} = \mathcal{L}_{km} p + \frac{\mathcal{S}_{km}}{p} + \mathcal{R}_{km}$, this is equivalent to: $Z_{km} i^m - \mathcal{L}_{km} i_0^m + \mathcal{S}_{km} q_0^m = e_k$. Usually, \mathcal{L}_{km} are the self inductance of meshes, $\mathcal{L}_{km}, k \neq m$ the mutual inductances between meshes. \mathcal{S}_{kk} the self elastance of meshes and $\mathcal{S}_{km}, k \neq m$ common elastances between meshes. \mathcal{R}_{kk} are the self

dissipations of meshes and $\mathcal{R}_{km}, k \neq m$ common dissipations between meshes. Note that for null initial conditions, Lagrange's equation can be written:

$$\left(\mathcal{L}_{km} p + \frac{S_{km}}{p} + \mathcal{R}_{km} \right) i^m = e_k \quad (48)$$

Second illustration on cellular spaces and second geometrization process

We take a look at the circuit presented in figure 10, including an operational amplifier.

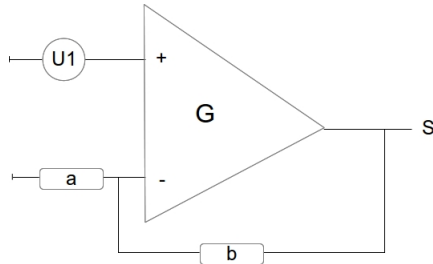


Figure 10: Circuit with operational amplifier

This circuit can be constructed starting from three elementary circuits. A first one (circuit 1) is the generator U_1 . The second one (circuit 2) is the amplifier seen as a coupled two edges circuit, with gain G as function of coupling. And the third circuit (circuit 3) is the output load giving the voltage S_4 . Figure 11 shows the set of these three circuits represented by four edges. So, in the edges space, this set can be compacted in a single tensor of properties Ze with all edges k of components z_k and the coupling G inside:

$$Ze = \begin{bmatrix} z_1 & 0 & 0 & 0 \\ 0 & z_2 & 0 & 0 \\ 0 & G & z_3 & 0 \\ 0 & 0 & 0 & z_4 \end{bmatrix} \quad (49)$$

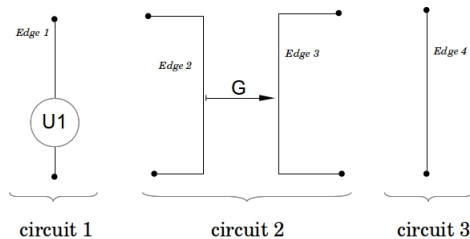


Figure 11: Three elementary circuits

To branch all these circuits together, we next define connectivity, saying that edges 1 and 2 belong to the same mesh 1, and edges 3 and 4 to the same mesh 2. By doing that, we mathematically translate the fact that edges 1 and 2 are soldered together and edges 3 and 4 together. The connectivity B is thus defined by (each row sends to the edges and each column to the meshes):

$$B = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} \quad (50)$$

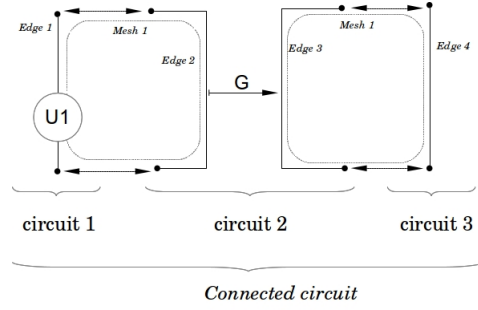


Figure 12: Connecting circuits

Figure 12 illustrates the mechanism.

The source (or command) covector is given by: $U_k = [U_1 \ 0 \ 0 \ 0]$. To obtain the mesh expression of the fundamental tensor we calculate $g = B^T Z e B$ which leads to:

$$z = \begin{bmatrix} z_1 + z_2 & 0 \\ G & z_3 + z_4 \end{bmatrix} \quad (51)$$

By the same transformation we obtain the source-command covector in the mesh space: $u = B^T U = [U_1 \ 0]$. Now to this system we want to add a control loop. For that we add a coupling function $-\alpha$ between edges 4 and 1 with:

$$\alpha = \frac{a}{a+b} \quad (52)$$

This results from adding to z the tensor of interactions

$$Q = \begin{bmatrix} 0 & -\alpha \\ 0 & 0 \end{bmatrix} \quad (53)$$

Then z becomes $g = z + Q$:

$$g = \begin{bmatrix} z_1 + z_2 & -\alpha \\ G & z_3 + z_4 \end{bmatrix} \quad (54)$$

With the choices $z_1 + z_2 = 1$ and $z_3 + z_4 = 1$, the system of equations $u_k = g_{km} x^m$ gives:

$$\begin{cases} U_1 = x^1 - \alpha x^2 \\ 0 = G x^1 + x^2 \end{cases} \quad (55)$$

For $S = -x^2$ we obtain easily $GU_1 = S(1 + \alpha G)$ which is the relation for the set-up given in figure 9. Convinced that our graph is equivalent to the circuit in figure 9, we wonder what could be a way to make g symmetric? To do that we calculate the derivative of each equation of (49) depending on x^1 then x^2 . It gives two vectors:

$$\mathbf{b}_1 = \begin{bmatrix} 1 \\ G \end{bmatrix} \quad \mathbf{b}_2 = \begin{bmatrix} -\alpha \\ 1 \end{bmatrix} \quad (56)$$

In this simple case, the jacobian matrix \mathcal{J} is equal to g and so, noting Γ the transposed matrix of \mathcal{J} (49) is equivalent to $\Gamma_{mk} u_k = \Gamma_{mk} \mathcal{J}_{kn} x^n$ with $\mathcal{G}_{mn} = \Gamma_{mk} \mathcal{J}_{kn}$, we obtain:

$$\mathcal{G} = \begin{bmatrix} 1 + G^2 & G - \alpha \\ G - \alpha & 1 + \alpha^2 \end{bmatrix} \quad (57)$$

which is a symmetric metric. Having this symmetric tensor, eigenvalues can be found, and the system of equations can be incorporated in a riemannian approach. The system of equations for the amplifier becomes (the source-command vector is $\Gamma u = [U_1 \ -\alpha U_1]$):

$$\begin{cases} U_1 = (1 + G^2) x^1 + (G - \alpha) x^2 \\ -\alpha U_1 = (G - \alpha) x^1 + (1 + \alpha^2) x^2 \end{cases} \quad (58)$$

If we calculate the determinant of this system we obtain $\Delta = (1 + \alpha G)^2$. The solution for x^2 is then given by:

$$x^2 = \frac{-U_1 G (1 + \alpha G)}{\Delta} \quad (59)$$

which is perfectly identical to the solution found previously. This new expression of the system is identical to the previous one but the dimensions of the metric is the square of the dimension of Kron's fundamental tensor.

Network evolution: third mathematical considerations

The properties of edges or mesh can change depending on functions set by parameters [1]. But topology also can change: edges or meshes can disappear, or sources and commands can change, etc. In general, parameters or variables in various matrices can change under the effect of the system users' decision. These changes come from the system life, including the modelled machine and its user. The idea is to translate the system life through the material transformations guided by the user's decisions. This can be reached by applying transformations to each component of the manifold \mathcal{M} : the source - command covector u_k or the metric g_{km} , the human actions being themselves guided by the material performances and their outputs. Finally, a new step consists in identifying the system through various objects involved in the manifold \mathcal{M} , then being able to apply transformations to these objects depending on users' psychological profiles and decisions. The tenfolds were created by the author in order to make this.

Transformations

Starting from an initial definition of a manifold \mathcal{M}_0 , we can change this definition by acting on its topology C (represented by the connectivity matrix); its commands (sources, etc.) T or its fundamental tensor (we call it G). Knowing that we can study various transformations that change the components of C , T or G , if Λ is a transformation matrix, the operations to make for the previous transformations can be summed up as [20]:

$$\Lambda_\mu^\nu C_\nu^\alpha \quad \Lambda_a^b T_b \quad \Lambda_\alpha^b G_{bf} \Lambda_\beta^f \quad (60)$$

Tenfold

An application A_e can be created giving each manifold a "tenfold". A tenfold is a simple list that includes the three major objects (C, T, G) . The name of tenfold comes from the name tensor and from the name of multi-sheet. If we note \check{u} a tenfold, by definition:

$$\check{u} = (C, T, G) \quad (61)$$

meaning that $\mathcal{M} \xrightarrow{A_e} \check{u}$ or $\mathcal{M} \xleftarrow{A_e^{-1}} \check{u}$ with the assumption that interval of values $I(u_k \in \check{u})$ for each variables are specified in association with the tenfold.

To specify that a tenfold has a single definition in a given phase of the system life, an index can be used in order to detail the current phase definition. For example in an initial state: $\check{u}_0 = (C, T, G)$. After some time duration, the definition can evolve in: $\check{u}_0 = (C, T', G)$ or other transformations.

Tenfold algebra

Once we have defined tenfolds and transformers, we can define a little algebra to give rules of transformations applied to tenfolds. Noting \hat{t}_x some affine transformer made of three transformation matrix component:

$$\hat{t}_x = \left(\Lambda_1, \Lambda_2, \left\{ \begin{array}{c} \Lambda_3 \\ \epsilon_3 \end{array} \right\} \right) \quad (62)$$

we define the rule of applying \hat{t}_x on the tenfold \check{u}_0 :

$$\begin{aligned}\hat{t}_x \cdot \check{u}_0 &= \left(\Lambda_1, \Lambda_2, \left\{ \begin{array}{c} \Lambda_3 \\ \epsilon_3 \end{array} \right\} \right) \cdot (C, T, G) = \dots \\ \dots &= (\Lambda_1 C, \Lambda_2 T, \Lambda_3^T G \Lambda_3 + \epsilon_3)\end{aligned}\tag{63}$$

It must be understood that other choices for the list of the objects included in the tenfold definition can be made, changing also the definition of the algebra product. The tenfold, transformer and algebra submitted here can be adapted depending on the problem [21].

Kuhn's tree and γ matrices for system evolution

Each state of a system in phase t can now be represented by a tenfold \check{u}_t . From one or various initial states, the system evolution at each phase of its life (the concept of phase can be linked with the real time or any other clock) can be followed using Kuhn's tree. Figure 13 shows one example of a system starting from a state \check{u}_0 and able to evolve into the states \check{u}_1 or \check{u}_2 .

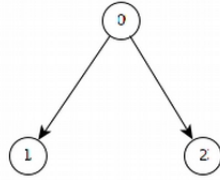


Figure 13: Kuhn's tree

The question is now to create the simplest possible mathematical context in order to theorize Kuhn's processes used here. Inspired from Markov's process of n-order, we can create a vector that memorizes all the states then to apply what we have called a γ matrix to change its components. For example, the previous tree can be synthesized in one initial step by the material vector \mathbf{MV}_0 given by:

$$\mathbf{MV}_0 = \begin{bmatrix} \check{u}_0 \\ 0 \\ 0 \end{bmatrix}\tag{64}$$

Now in the next step, the system can be in either \check{u}_1 or \check{u}_2 states through the γ_{10} matrix:

$$\gamma_{10} = \begin{bmatrix} 0 & 0 & 0 \\ \hat{t}_{10} & 0 & 0 \\ \hat{t}_{20} & 0 & 0 \end{bmatrix}\tag{65}$$

With $\check{u}_1 = \hat{t}_{10} \cdot \check{u}_0$ and $\check{u}_2 = \hat{t}_{20} \cdot \check{u}_0$, this leads to:

$$\gamma_{10} \cdot \mathbf{MV}_0 = \mathbf{MV}_1 = \begin{bmatrix} 0 \\ \check{u}_1 \\ \check{u}_2 \end{bmatrix}\tag{66}$$

The γ matrix that wears all the transformers is the fundamental matrix of the system evolution mechanism. It gives all the definition of this evolution. Two approaches can so be used: define various γ matrices and then imagine the evolution of the system for its complete life until step n :

$$\mathbf{MV}_n = \gamma_{n n-1} \gamma_{n-1 n-2} \dots \gamma_{10} \mathbf{MV}_0\tag{67}$$

or create and delete matrices which can be defined in order to increase or decrease the dimensions of the vector \mathbf{MV} and of the matrix γ following the system evolution in real time [1].

Game theory and human factor

We have defined the machine mathematically. We want to add the human influence on the way the machine works as well as the machine's impact on the human behaviour (note that it could be animal-machine system in general). Our approach is to use game theory [22] to add the human factor side. The principle is simple: each transition from one machine state to another can be associated with a Bayesian probability. This Bayesian probability depends on human choice, of environmental impact and previous state of the machine, etc.

Technically, we must create mathematical objects as previously to incorporate these notions. An information vector is created \mathbf{IV} . It has the probabilities attached to each machine state, starting with the initial information 1. For our previous example in the initial step, it gives:

$$\mathbf{IV}_0 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad (68)$$

A $\tilde{\gamma}$ matrix which is a Markovian one applies the same process to the information vector \mathbf{IV} where all components p_{ji} are Bayesian probabilities going from state \check{u}_i to state \check{u}_j . In our case, $\tilde{\gamma}_{10}$ is:

$$\tilde{\gamma}_{10} = \begin{bmatrix} 0 & 0 & 0 \\ p_{10} & 0 & 0 \\ p_{20} & 0 & 0 \end{bmatrix} \quad (69)$$

Giving:

$$\mathbf{IV}_1 = \tilde{\gamma}_{10}\mathbf{IV}_0 \quad (70)$$

Perfectly symmetric to the machine evolution, the information process (or "psychological" process, or "psychical" process) is given by:

$$\mathbf{IV}_n = \tilde{\gamma}_{n\ n-1} \dots \tilde{\gamma}_{10}\mathbf{IV}_0 \quad (71)$$

The same two approaches exist to describe the whole system life, as we did for the machine part.

The rules of the game are defined for each evolution. It means defining the assumptions: who are the actors, what are the rules, who plays, in what time, what are the payoffs, etc. This must be done before any computation.

Human machine cross talk

Some objectives fix the target that the human - machine system (HMS) wants to reach. In each step, the observables indicate if the objective is reached. If not, the HMS evolves to increase its performance. Depending on the human psychological profile, transformations are different. That is why the probabilities associated to each transformation are Bayesian. For example to go from state 1 to 2:

$$p_{21} = P(\mathbf{MV}_2 | \theta \in \mathbf{MV}_1(\check{u}_x)) \quad (72)$$

θ being the observable. But the machine itself can act depending on the environment, or the user, or its own reliability, etc. Finally:

$$p_{21} = P(\hat{t}_{21}) P(\mathbf{MV}_2 | \theta \in \mathbf{MV}_1(\check{u}_x)) \quad (73)$$

All $P(\mathbf{MV}_j | \theta \in \mathbf{MV}_i(\check{u}_x))$ translates the man choice guided by the machine response, while all $P(\hat{t}_{ji})$ translates the machine action guided by the human decision. The system is completely coupled for a given psychological profile [23]. Note that the objectives can be associated with payoff functions in game theory.

Choices - payoff representation

For complex trees, it becomes difficult to understand the behaviours. In order to give the simplest interpretations of evolutions, a solution consists in looking at the various trajectories in the evolution tree (ET: Kuhn's tree showing the HMS possible evolutions). Following a sequence of states in the ET, the payoff $\$k$ function of a trajectory with last state \hat{u}_k can be estimated by:

$$\$(\hat{t}_{k k-1}, \theta_{(k)}) = \theta_{(k)} \prod_0^k p_{k k-1} = \mathbf{IV}^{(k)} \theta_{(k)} \quad (74)$$

$\theta_{(k)}$ being the performance for the state k and $p_{k k-1}$ the probability to reach the state from the previous one in the considered way.

This trajectory can be visualized in a graph where axes are possible groups of transformations and coordinates are the probabilities to use these groups of transformations. One axe is added: the payoff function. Figure 14 shows a tree with three possible trajectories. For each of them, payoffs'

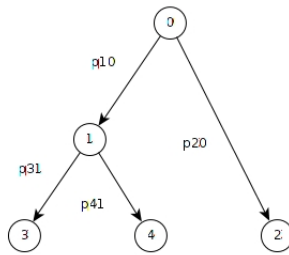


Figure 14: Trajectories

are: $\$2 = p_{20}\theta_2$, $\$3 = p_{10}p_{31}\theta_3$ and $\$4 = p_{10}p_{41}\theta_3$. If \hat{t}_{10} and \hat{t}_{41} belongs to the same group GR_1 and \hat{t}_{31} and \hat{t}_{20} to a second one GR_2 , the three trajectories can be projected in the graph presented in figure 15. When many curves are drawn, clear behaviours appear through this representation

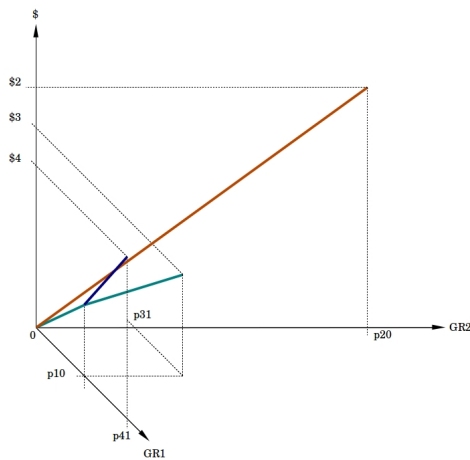


Figure 15: Payoff - transformations representation

as "Nash's horizon", etc. [1]. This last operation in the submitted theory is an important key to conduct complex systems analysis.

Applications

Some applications can use the proposed xTAN technique. Applications were used in electromagnetic compatibility (EMC) [24] [25] [26]. The exercise is difficult, in correspondence with the complexity of the problems considered. But very efficient results were found. It is clear that showing these applications in detail cannot be part of this article, needing one article to present a single case. For example in [27] the relation between customers and suppliers was studied. The psychological profile of the customer was one of the assumptions. It is sometimes quite easy to define this kind of profiles. Here, limits in competence were evaluated. For a customer who is not a specialist, the probability of acceptance of a test result was 1/2. In other cases, the probability of acceptance was a function of the average distance between the results and the standard limits (observable θ_k). The demonstration was made that the higher the competence of the customer, the higher the effort of the supplier to make high performance equipment. The payoff function was based on the supplier's effective cost for the equipment development.

Illustration

To illustrate the mechanisms, we imagine this situation inspired from the well-known prisoner dilemma in game theory [5]. Here we do not solve this problem for which solutions are known, but to better illustrate the technique, we show how it may be treated using xTAN. Two players are separated in cages. They know that to get out of the cage they must not activate a robot, but someone (the "environment" ξ) tries to give them the doubt saying that they should activate the robot for to be free. He adds that if one gamer activates the robot and the other gamer does not activated the robot, the first gamer will be free in one half day and the other will stay in jail for two days. Once in the cages, they cannot speak together, nor see each other. So all the strategy is based on faith. The game accepts one shot only. Depending on the command applied to the robot, the gamer can stay half a day, one day or two days in jail. If neither of the gamers activates the robot, they are free. The robot controls the jail openers. The payoff matrix of the game is given in figure 16.

	Gamer 1 activation	Gamer 1 Nothing done
Gamer 2 activation	Gamer 1 : 1 Gamer 2 : 1	Gamer 1 : 2 Gamer 2 : 0,5
Gamer 2 Nothing done	Gamer 1 : 0,5 Gamer 2 : 2	Gamer 1 : 0 Gamer 2 : 0

Figure 16: Payoff matrix for the prisoner game

The diagram associated with this system is shown in figure 17. Two meshes are used to represent the first gamer's actions, and one mesh for the second gamer. Depending on the gamers' spirit (translated in the fundamental tensor) and for the solicitation coming from the environment ξ , the system gives through the mesh currents s^1 and s^2 the number of jail days for each gamer. A fourth mesh is added to compute the system gain $\$_{\theta}$ given by: $\$_{\theta} = 2.5 - s^1 - s^2$.

In this graph two cords exist between meshes 1 and 2. They are directly linked with the trust each player gives to the other as we will see. The fundamental tensor at the initial state of this graph is ($\alpha = \beta = 1$):

$$z = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & -1 & -1 & 1 \end{bmatrix} \quad (75)$$

for a command vector $u = [x \ x \ -1 \ 2.5]$. Before entering the game, the players trust each other. So their coupling coefficients α and β are equal to 1. For the solicitation coming from the

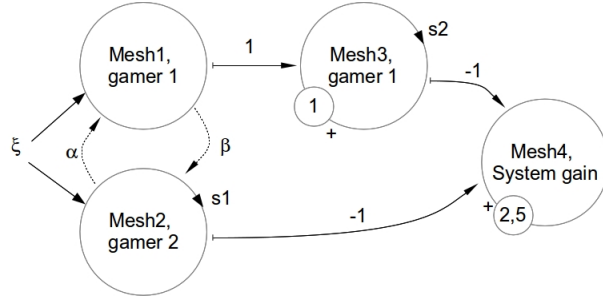


Figure 17: Diagram for the prisoner game

environment, as the game has not yet begun we don't know its values and we can note them "x" ($\xi = x$). The transformers will act on the first quadrant of z only because the last two meshes are only here to compute the gain but not to model the process involved. Here we do not want to change the topology. The tenfold can be reduced to the components T and G , giving:

$$\check{u}_0 = (u, z) \quad (76)$$

Once the game has started ($\xi = 1$), if both players refuse to activate the robot (non activation: (N, N)), it means that they completely trust each other and change nothing on the initial state. Note that for this tensor, the equation $u_k = z_{km} t^m$ leads to $s^1 = s^2 = 0$ and $\$_{\theta} = 2.5$ with $u_1 = u_2 = 1$. The transformer is equal to (always modifying only the first quadrant):

$$\hat{t}_1 = \left(\Lambda_2 = \begin{bmatrix} \frac{1}{x} & 0 \\ 0 & \frac{1}{x} \end{bmatrix}, \Lambda_3 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \epsilon_3 = [0] \right) \quad (77)$$

For the second case (N, A) (A for activation) we obtain:

$$\hat{t}_2 = \left(\Lambda_2 = \begin{bmatrix} \frac{1}{x} & 0 \\ 0 & \frac{1}{x} \end{bmatrix}, \Lambda_3 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \epsilon_3 = \begin{bmatrix} 0.3 & 0 \\ \frac{0.3}{2} & 0 \end{bmatrix} \right) \quad (78)$$

For the case (A, N) :

$$\hat{t}_3 = \left(\Lambda_2 = \begin{bmatrix} \frac{1}{x} & 0 \\ 0 & \frac{1}{x} \end{bmatrix}, \Lambda_3 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \epsilon_3 = \begin{bmatrix} 0 & -0.25 \\ 0 & 0.5 \end{bmatrix} \right) \quad (79)$$

And for the last case (A, A) :

$$\hat{t}_4 = \left(\Lambda_2 = \begin{bmatrix} \frac{1}{x} & 0 \\ 0 & \frac{1}{x} \end{bmatrix}, \Lambda_3 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \epsilon_3 = \begin{bmatrix} 0.3 & 0 \\ \frac{0.3}{2} & 0 \end{bmatrix} \right) \quad (80)$$

In each case k the transformation is applied through:

$$\check{u}_k = \hat{t}_k \cdot \check{u}_0 = (\Lambda_2 u, \Lambda_3^T z \Lambda_3 + \epsilon_3) \quad (81)$$

and gives the output specified in the payoff matrix given in figure 16. Note that in the case (A, A) , no trust at all exists between the players and the coupling component of z is zero. The first four quadrant matrix of z after transformations is:

$$\left(\begin{array}{l} (N, N) \rightarrow \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \\ (N, A) \rightarrow \begin{bmatrix} 0.3 & 0 \\ \frac{1}{6} & 1 \end{bmatrix} \\ (A, N) \rightarrow \begin{bmatrix} 1 & -0.25 \\ 0 & 0.5 \end{bmatrix} \\ (A, A) \rightarrow \begin{bmatrix} 0.5 & 0 \\ 0 & 1 \end{bmatrix} \end{array} \right) \quad (82)$$

When a player chooses "N", he allocates some faith to his partner. In this case, the coupling coefficient is not zero. This is a guide to understand the results in z . After what, values are obtained for the choosen topology. As there is only one shot in the game, the γ matrix is given by:

$$\gamma_{10} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ \hat{t}_{10} & 0 & 0 & 0 & 0 \\ \hat{t}_{20} & 0 & 0 & 0 & 0 \\ \hat{t}_{30} & 0 & 0 & 0 & 0 \\ \hat{t}_{40} & 0 & 0 & 0 & 0 \end{bmatrix} \quad (83)$$

equation of the game being similar to (66). The four gains are: 2.5, 0, 0, 0.5. Now we have to define the probabilities associated with each transformation.

We understand that here, we clearly touch the more complex side of the method. The probabilities are linked with the self-confidence that each player has, with the credibility they allocate to the third player (the environment) and the faith they have in their partner. Game theory in general considers rational players, i.e. players that don't have other things in mind than the pure objectives. But this is not real life. And to control complex systems, we must take into account real life. The solution consists in working with ergonomists who make statistics with well-known population and well-known psychological profiles. These difficulties were noticed even in the prisoner dilemma [28]. But another very interesting point for system studies is that even if the probabilities are not completely correct in comparison to measurements made with populations, decisions deduced from the xTAN computation can lead to adequate solutions. The Markov matrix has the following structure:

$$\tilde{\gamma}_{10} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ p_{10} & 0 & 0 & 0 & 0 \\ p_{20} & 0 & 0 & 0 & 0 \\ p_{30} & 0 & 0 & 0 & 0 \\ p_{40} & 0 & 0 & 0 & 0 \end{bmatrix} \quad (84)$$

If we can imagine that ergonomists have made tests, and results are those given in figure 18 for three kinds of psychological profile populations, using the results, we can make a graph giving the gains (x100) depending on the four possible transformations for the system: (N, N) , (N, A) , (A, N) , (A, A) . This graph is shown in figure 19.

Profile	Rational	doubling	afraid
case			
N,N	90,00%	70,00%	50,00%
A,N	4,00%	12,00%	20,00%
N,A	4,00%	12,00%	20,00%
A,A	2,00%	6,00%	10,00%

Figure 18: Probability values

This representation shows that the highest gains (for the given definition) are obtained for rational persons in case (N, N) , case where they cannot be influenced by another one. But for some population, some intermediate gains are obtained. Note that even for the rational population, persons of case (A, A) can exist, for example two persons making an error in the understanding of the rules or that activate the robot by accident. This kind of minority behaviours are important for risk prediction. That is a particularity of human factor inclusion in system analysis.

Conclusion

It seems to us that the xTAN method may offer various interesting aspects for the automatic community. Compared to classical bond-graphs or classical block-diagrams, it also gives models

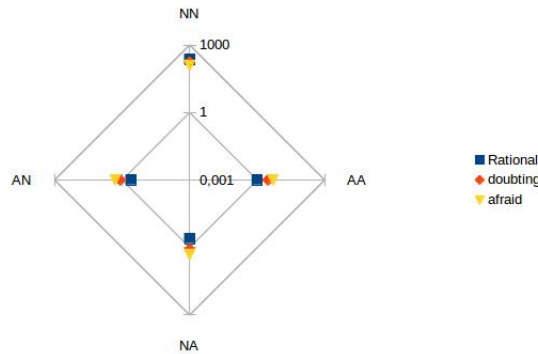


Figure 19: Decision - gains curves

incorporated in graphs that can be enriched as will. Links between the drawn graph and the fundamental tensor of Kron gives a gateway to differential geometry. It means that automatic problems can be seen under the geometrical point of view, even if many applications have to be developed. As it has made its proof in complex system analysis for the electromagnetic compatibility, we can trust that it may lead to new visions in automatic, inspired also by those already have been made by Kron in similar cases [29].

But the originality of our proposal reside also in coupling Kron's method with game theory, taking into account human factor. The coupled theory remains to be used on many case, in order to demonstrate its efficiency, but some first cases have already shown the potential of the approach. Future development will consist in using the xTAN method on quite classical but already quite complex problems in automatic, to see if the graph can be easier than block-diagrams to describe these cases, and to solve them after. Looking at block-diagrams and graphs we see that the principle of each drawing is a little reversed. In graphs, the "blocks" are more often associated with mesh while blocks of the block-diagrams representation are associated with cords. The interpretation of the graph is immediate. There is no need to move connexions for example. The system of equations linked to the graph is directly obtained without needing any operation to simplify the graph.

We hope that these possibilities will generate interesting new approaches in automatic, perhaps in the graph construction and exploitation, but at least in the game theory coupling process to incorporate human factor.

References

1. MAURICE, Olivier. Introduction d'une théorie des jeux dans des topologies dynamiques. 2013. Ph.D. thesis. Limoges university, France, 2013.
2. MAURICE, Olivier, REINEIX, Alain, DURAND, Ph, et al. On mathematical definition of chords between networks. In : European electromagnetics, EuroEM 2012. 2012.
3. Durand, Philippe, et al. "Networks, Topology and Interaction Principle Implemented in the Kron's Method." *Journal of Mathematics and System Science* 4.6 (2014).
4. Korchounov, Y., and Vladimir Kotliar. "Fondements mathématiques de la cybernétique." (1975).
5. Aghion, Philippe, and Jean Tirole. "Formal and real authority in organizations." *Journal of political economy* (1997): 1-29.
6. Maurice, Olivier. *La compatibilité électromagnétique des systèmes complexes*. Hermes science publ., 2007.

7. Maurice, Olivier, Philippe Durand, and Alain Reineix. "Methodology of Network Analysis Using Xtan The Second Geometrization Approach." 5th International Conference on Computational Mathematics, Computational Geometry Statistics. 2016.
8. Poincaré, Henri. *Analysis situs*. na, 1895.
9. Olivier Maurice. *Caractères remarquables de la méthode de Kron*. 2016. [jhal-01282162](#);
10. Olivier Maurice, Philippe Durand. *Modélisation des systèmes complexes. : Présentation des grands principes de la méthode xTAN*. Présentation effectuée lors de la réunion du groupe CESIR de l'AFSCET à l'ESIGELEC, le 19 décembre. 2015. [jhal-01246740](#);
11. Troyanov, Marc. *Cours de géométrie*. PPUR presses polytechniques, 2002.
12. Kron, Gabriel. "Tensor analysis of networks." New York (1939).
13. MAURICE, Olivier. "Analyse tensorielle des réseaux appliquée à la compatibilité électromagnétique de l'électronique de puissance." *La Revue 3 EI* 65 (2011): 36-43.
14. Roth, J. Paul. "An application of algebraic topology to numerical analysis: On the existence of a solution to the network problem." *Proceedings of the National Academy of Sciences* 41.7 (1955): 518-521.
15. Roth, J. Paul. "The validity of Kron's method of tearing." *Proceedings of the National Academy of Sciences of the United States of America* 41.8 (1955): 599.
16. Roth, J. Paul. "An application of algebraic topology: Kron's method of tearing." *Quarterly Of applied mathematics* 17.1 (1959): 1-24.
17. Hoffmann, Banesh. "Kron's non-Riemannian electrodynamics." *Reviews of modern physics* 21.3 (1949): 535.
18. Olivier Maurice. *Pistes pour l'application de la notion de variété aux espaces de Kron, et la justification de la pertinence d'emploi du concept de tenseurs par Kron*. Document rapport de réflexions en cours sur le sujet menées par l'auteur. 2014. [jhal-01077631](#);
19. Olivier Maurice. *Kron's formalism as best candidate for a second geometrization*. 2014. [jhal-01079386](#);
20. Denis-Papin, Maurice, and A. N. Kaufman. "Cours de calcul tensoriel appliqué." (1966).
21. Olivier Maurice, P Durand, A Reineix, François Dubois. *PROPOSITION D'UNE THEORIE POUR LA CEM*. CEM2014, Jul 2014, Clermont Ferrand, France. [jhal-01254213](#);
22. Fudenberg, Drew, and Jean Tirole. "Game theory, 1991." Cambridge, Massachusetts 393 (1991).
23. Marchais, Pierre. *Les infrastructures du trouble mental: renouvellement des modes de pensée en psychiatrie*. Editions L'Harmattan, 2015.
24. Maurice, Olivier, Alain Reineix, and Sébastien Lalléchère. "Electromagnetic game modeling through Tensor Analysis of Networks and Game Theory." *IOP Conference Series: Materials Science and Engineering*. Vol. 67. No. 1. IOP Publishing, 2014.
25. Olivier Maurice, Alain Reineix. *Proposition d'une méthode xTAN pour modéliser les processus complexes*. 2012. [jhal-00695458v2](#);
26. Olivier Maurice. *Eléments de systématique*, IRSEEM, France. 2015. [jcel-01299650](#);

27. Reineix, Alain, and Olivier Maurice. "Modélisation de la relation client fournisseur en CEM." 16ème Colloque International et Exposition sur la Compatibilité Electromagnétique-CEM 2012. 2012.
28. Eber, Nicolas. Théorie des jeux. Paris: Dunod, 2007.
29. Kron, Gabriel. "Quasi-Holonomic Dynamical Systems." Journal of Applied