POWER CHOPPER MODELLING USING THE KRON’S METHOD

Philippe Durand
M2N
CNAM
292 rue Saint Martin 75141
Paris, France
e-mail: philippe.durand@cnam.fr

Folla Boussandel and Olivier Maurice
EMC Laboratory
GERAC
3 av. Jean d’Alembert 78190
Trappes, France
e-mail: olivier.maurice@gerac.com
e-mail: folla.boussandel@gerac.com

ABSTRACT
The aim of this article is to present the main aspect of an numerical implementation under the Kron’s method of a power chopper modelling developed by Denis Labrousse, Bertrand Revol and Francois Costa from SATIE laboratory, France. We name the model LRC. The Kron’s method is based on a tensorial analysis of networks (TAN). The aim of our study is to show that this method can be very efficient to compute serial chopper with one inverter arm through the LRC modelling, including a command law \( f_{sw}(t) \) applied on both switchers. Once this work done, the TAN will allow us to easily add the other elements of the system: cables, battery, electrical machine.

KEY WORDS
Mechatronics, Industrial engineering, Kron’s method, tensorial analysis.

1. Introduction
We present the LRC modelling based on a voltage and current report depending on the switchers state[1]. After what we present the Kron’s method[2] in a complete space[3], then the LRC implementation, with some self choices we made to construct the previous generators needed for LRC. This implies topological choices for the various components involved. Finally we present the programming of the chopper using Python’s language. We then conclude on performance and future works.

2. LRC modelling
The model is made of two circuits: a first circuit modelling the input of a chopper converter and a second circuit modelling the output of the chopper. On entry, we want to postponed the storage of energy coming from the power supply inside the input capacitor and the discharge in the phase where the top floor of the chopper is opened and the bottom one is closed. On output we want modelling the potential reported on the load (top floor chopper switch closed and bottom one opened ). Figure 1 presents the cop-
generalize this relation by: \([V] + [E] = [Z][I]\), noting that we can define \(V\) as a node-pair vector in a node-pair space:

\[
V = \begin{bmatrix}
V1 \\
V2 \\
. \\
. \\
Vn
\end{bmatrix}
\] (1)

such as: \(V = C^t \ast v^t\), noting that \(C\) is a connection matrix and \(v\) is the vector of nodes pair in the space of branches. We now detail each object \(v, E, Z\) and \(I\).

3.1 Vector \(v\)

\[
v = \begin{bmatrix}
v1 \\
v1 \\
. \\
. \\
v_n
\end{bmatrix}
\] (2)

By the same way that we define a node pair space, we can define a mesh space, starting form the relation between the branch current vector \(i_b\) and the mesh current covector \(i_m\). We have: \(i_b = C i_m\). This gives the connection matrix used to go through both spaces, branch and mesh ones.

3.2 Covector \(E\)

The sources in the mesh space are included in a covector \(E\):

\[
E = \begin{bmatrix}
E1 \\
E2 \\
. \\
. \\
E_n
\end{bmatrix}
\] (3)

such as: \(E = C^t \ast e^t\), noting that \(C\) is the connection matrix and \(e\) is the source covector of branches in the space of branches,

\[
e = \begin{bmatrix}
e1 \\
e1 \\
. \\
. \\
e_n
\end{bmatrix}
\] (4)

3.3 Twice covariant tensor \(Z\)

From the invariant theory and tensorial algebra, we can define a metric which in our case is the \(Z\) impedance matrix, existing both in the branch and the mesh space. Using the metric \(Z\), and only by this, it is possible to transform a vector in a covector (and the inverse using the inverse metric), i.e. to go form a current to a voltage.

3.4 Vector \(I\)

I is a vector that comes from the summation of vectors \(i_c\) of node pairs and \(i_m\) of meshes. The node pair current vector is deduced form the current sources through the connection matrix \(K\): \([i_c] = [K] \ast [j]\). \(K\) is purely diagonal with 1 when it exists a current source on the corresponding node pair, and 0 elsewhere:

\[
K = \begin{bmatrix}
1 & 0 & . & . & 0 \\
0 & 1 & . & . & . \\
. & . & . & . & . \\
. & . & . & . & . \\
0 & . & . & . & 1
\end{bmatrix}
\] (5)

The equation obtained is described in well named complete space, exploiting both nodes pair and meshes spaces.

4. Kron’s method implementation of LRC modelling

To implement LRC, we first define a command law, after an equivalent circuit of our chopper then we justify the topology chosen.

4.1 Command law \(fsw(t)\)

The command law is basically a succession of 1 and 0 making the switches open or closed. But this kind of function is not realistic, because real generators have always delays for rise or fall time. That’s why we decide to create a fast gaussian realistic representation of such signal. The function is given by: \(f_{sw}(t) = A \ast \exp\left(\frac{t-x_0}{\sigma}\right)^{10}\), noting that \(x_0\) is the gaussian center location and \(\sigma\) its duration. \(A\) is the gaussian amplitude. Our Gaussian has three parts: a null part, a flat part and an exponential part expressed in Figure 3 by the small delay \(\Delta t\). This delay has been introduced so that we can easily solve our equations with the finite difference method in temporal domain and minimizing the numerical computation errors.

Figure 3: Gaussian signal for command law
4.2 Equivalent circuit of our system

Figure 4 shows the equivalent circuit of our system with the voltage node pair source and the controlled current source. Our LRC modelling is separated into two networks such that each one is controlled by the other. The second network is powered by the voltage $V_{dc}(t)$ reported from the first network, and the load current of the second network $i_s$ is injected in the first network depending on the command law.

The second network includes a generator $E_2$, given by: $E_2 = V_{dc} \times f_{sw}$.

4.3 Topology for the system

The visible elements in the graph given Figure 5 are the topological following character:

- 4 physical nodes $n_1,...,n_4$ ($\rightarrow N = 4$)
- 5 branches $b_1,...,b_5$ ($\rightarrow B = 5$)
- 3 meshes $m_1,m_2,m_3$ ($\rightarrow M = 3$)
- 2 networks $R_1, R_2$ ($\rightarrow R = 2$)
- 2 nodes pair ($\rightarrow P = 2$)

We construct by this return, the first couple $P_1$ who will wear the current source $J_1$, and will be in final, our current injected in the first network coming from the second network.

We verify the relationship for node pair: $P = N - R = 4 - 2 = 2$ and meshes: $M = B - N + R = 5 - 4 + 2 = 3$. As in our first Network, we choosing arbitrarily on our second network the Node $n_3$, as reference from depart. We depart of this Node worm Node $n_4$, we have an return from Node $n_4$ to Node $n_3$, we construct with this return, the second couple "P1", who will wear normally the current source $J_2$, but all along our study we suppose that $J_2$ is null, because it is rattaché to a branch which comport a current source.

we will have a single current source $J_1$, of the first network. For the second network, we can choose another topology, different than that above, we do not change the choice of current and couple, but we can change the choice of the meshes. But the computations give the same results in both case due to the invariant theorem. In all our calculs, we choose the topology presented figure 6.

we choosing arbitrarily on our first network, the node 1 as an initial reference, we depart of this Node worm node 2, we have an return of Node 2 to Node 1.
4.4 Connection matrix

According to the complete space of first and second network look at Figure 6 and Figure 7, according to the topological character determined before, we can determine the connection matrix linking meshes, nodes pair and branches currents:

\[
\begin{bmatrix}
  i_1 \\
i_2 \\
i_3 \\
i_4 \\
i_5 \\
\end{bmatrix} = \begin{bmatrix}
  1 & 0 & 0 & 0 & 0 \\
  1 & 1 & 0 & 0 & 0 \\
  0 & 0 & 0 & 1 & 0 \\
  0 & 0 & 0 & 0 & 1 \\
\end{bmatrix} \begin{bmatrix}
  i_{m1} \\
i_{m2} \\
i_3 \\
j_1 \\
\end{bmatrix}
\]

\[i(t) = \begin{bmatrix}
  i_{m1} \\
i_{m2} \\
i_3 \\
j_1 \\
\end{bmatrix}
\]

(6)

4.5 Impedance tensor

The impedances tensor \( z \) in the space of the branches is:

\[
z = \begin{bmatrix}
  R_e + L_e * p & 0 & 0 & 0 & 0 \\
  0 & \frac{1}{C_{ss}*p} & 0 & 0 & 0 \\
  0 & 0 & R_s + L_s * p & 0 & 0 \\
  0 & 0 & 0 & 0 & 1 \\
  0 & 0 & 0 & 0 & R_{dr} \\
\end{bmatrix}
\]

Applying the bilinear transformation given by: \( Z = C' \cdot z \cdot C \), we obtain the impedance tensor \( Z \) in the mesh space:

\[
Z = \begin{bmatrix}
  a & 0 & 0 & -\frac{1}{C_{ss}*p} \\
  0 & b & 0 & \frac{1}{C_{ss}*p} \\
  0 & \frac{1}{C_{ss}*p} & c & 0 \\
  -\frac{1}{C_{ss}*p} & 0 & 0 & d \\
\end{bmatrix}
\]

(7)

with \( a = R_e + L_e * p + \frac{1}{C_{ss}*p}, b = R_s + L_s * p + \frac{1}{C_{ss}*p}, c = R_{dr} + \frac{1}{C_{ss}*p}, d = \frac{1}{C_{ss}*p} \).

4.6 Source vector

The source covector in the branch space is given by:

\[
s = \begin{bmatrix}
  E & 0 & f_{sw} * V_{dc} & 0 & 0 \\
\end{bmatrix}
\]

(9)

After transformation we obtain the source covector in the mesh space:

\[
S = C' \ast e^t = \begin{bmatrix}
  E \\
  -f_{sw} \ast V_{dc} \\
  0 \\
\end{bmatrix}
\]

(10)

4.7 Voltage covector in the branch space

The voltage covector in the branches space is given by:

\[
v = \begin{bmatrix}
  0 & V_{dc} & 0 & 0 & 0 \\
\end{bmatrix}
\]

(11)

Following similar transformation gives:

\[
V = C' \ast v^t = \begin{bmatrix}
  V_{dc} \\
  0 \\
  -V_{dc} \\
\end{bmatrix}
\]

(12)

5. Time domain resolution of the tensorial equation

We want to resolve the integrodifferential equation:

\[
\begin{bmatrix}
  E \\
  \end{bmatrix} + \begin{bmatrix}
  V \\
  \end{bmatrix} = \begin{bmatrix}
  Z \\
\end{bmatrix} \begin{bmatrix}
  I \\
  \end{bmatrix}
\]

(13)

Using the method of finite difference, time domain[4], we obtain the following equations:

\[
\begin{align*}
E(t) + V_{dc}(t) &= R_e i_{m1}(t) + \frac{dt}{dt} (i_{m1}(t) - i_{m1}(t - 1)) + \frac{dt}{dt} \sum_{i=1}^{N} i_{m1}(t) - \frac{dt}{dt} \sum_{j=1}^{N} J_1(t) \\
&- f_{sw}(t) V_{dc}(t) = \frac{dt}{dt} \sum_{i=1}^{N} i_{m1}(t) + \frac{dt}{dt} \sum_{i=1}^{N} i_{m2}(t) + \frac{dt}{dt} \sum_{i=1}^{N} i_{m2}(t) + \frac{dt}{dt} \sum_{i=1}^{N} J_1(t) \\
&- f_{sw}(t) V_{dc}(t) = R_{dr} i_{m3}(t) + \frac{dt}{dt} \sum_{i=1}^{N} i_{m3}(t) + \frac{dt}{dt} \sum_{i=1}^{N} J_1(t) \\
-V_{dc}(t) &= -\frac{dt}{dt} \sum_{i=1}^{N} i_{m1}(t) + \frac{dt}{dt} \sum_{i=1}^{N} J_1(t)
\end{align*}
\]

(14)

We must now set all the terms before the running time of the source side:
\[
E(t) + V_{dc}(t) + \frac{\delta}{\delta t} i_m(t - 1) - \frac{\delta}{\delta t} \sum_{i=1}^{N-1} i_{m_i}(t) \\
+ \frac{\delta}{\delta t} \sum_{i=1}^{N-1} J_i(t) \\
= R_e \cdot i_{m_1}(t) + \frac{\delta}{\delta t} i_{m_1}(t) + \frac{\delta}{\delta t} i_{m_1}(t) + \frac{\delta}{\delta t} i_{m_1}(t) \\
-f_{sw}(t) V_{dc}(t) - \frac{\delta}{\delta t} \sum_{i=1}^{N-1} i_{m_1}(t) - \frac{\delta}{\delta t} \sum_{i=1}^{N-1} i_{m_2}(t) \\
+ \frac{\delta}{\delta t} (i_{m_2}(t - 1) \\
= \frac{\delta}{\delta t} i_{m_1}(t) + \frac{\delta}{\delta t} i_{m_2}(t) + \frac{\delta}{\delta t} i_{m_2}(t) + R_s i_{m_2}(t) \\
-f_{sw}(t) V_{dc}(t) - \frac{\delta}{\delta t} \sum_{i=1}^{N-1} i_{m_2}(t) \\
- \frac{\delta}{\delta t} \sum_{i=1}^{N-1} i_{m_3}(t) \\
= R_{dr} i_{m_3}(t) + \frac{\delta}{\delta t} i_{m_2}(t) + \frac{\delta}{\delta t} i_{m_3}(t) \\
-V_{dc}(t) + \frac{\delta}{\delta t} \sum_{i=1}^{N-1} i_{m_1}(t) - \frac{\delta}{\delta t} \sum_{i=1}^{N-1} J_i(t) \\
= - \frac{\delta}{\delta t} i_{m_1}(t) + \frac{\delta}{\delta t} J_i(t)
\]

Finally we obtain the two matrices \(W\) (impedance tensor matrix in the space of meshes) and \(T\) (sources covector in the space of meshes):

\[
T = \begin{bmatrix}
E(t) + \frac{\delta}{\delta t} i_{m_1}(t - 1) & \frac{\delta}{\delta t} \sum_{i=1}^{N-1} i_{m_1}(t) & \frac{\delta}{\delta t} \sum_{i=1}^{N-1} J_i(t) \\
-f_{sw}(t) V_{dc}(t) - \frac{\delta}{\delta t} \sum_{i=1}^{N-1} i_{m_2}(t) & \frac{\delta}{\delta t} \sum_{i=1}^{N-1} i_{m_2}(t) & \frac{\delta}{\delta t} \sum_{i=1}^{N-1} J_i(t) \\
-V_{dc}(t) + \frac{\delta}{\delta t} \sum_{i=1}^{N-1} i_{m_1}(t) & \frac{\delta}{\delta t} \sum_{i=1}^{N-1} i_{m_3}(t) & \frac{\delta}{\delta t} \sum_{i=1}^{N-1} J_i(t)
\end{bmatrix}
\]

\[
W = \begin{bmatrix}
\frac{L_e}{\delta t} + \frac{\delta}{\delta t} + R_e & 0 & 0 & -\frac{\delta}{\delta t} \\
0 & \frac{L_e}{\delta t} + \frac{\delta}{\delta t} + R_s & \frac{\delta}{\delta t} & 0 \\
-\frac{\delta}{\delta t} & \frac{\delta}{\delta t} & \frac{\delta}{\delta t} + R_{dr} & 0 \\
0 & 0 & 0 & \frac{\delta}{\delta t}
\end{bmatrix}
\]

The equation was implemented under a python program. It takes about two minutes without any problem of convergence, despite the fact that the numerical schematic used is here the simplest one. For example, Figure 9 shows the load current depending on the command law \(f_{sw}(t)\).

6. Conclusion

Kron’s method allowed to implement quite easily the MRC modelling. The results under python shown good stability and speed of computation. Next work will consist in adding the other elements of a typical power chain as cables, battery and motors. For these components, previous works and modelling give all the material to make these tasks quite easily. Comparison with experiments will validate the results and allow to increase the number of components to simulate near to real power systems. Its sure that the TAN gives all the mathematical tools to reach this ambitious objective.

References