

# Simulation of Arc Models with the Block Modelling Method

R. Thomas, D. Lahaye, C. Vuik, L. van der Sluis

*Abstract- Simulation of current interruption is currently performed with non-ideal switching devices for large power systems. Nevertheless, for small networks, non-ideal switching devices can be substituted by arc models. However, this substitution has a negative impact on the computation time. At the same time, these simulations are useful to design switchgear. Although these simulations are for large power systems cumbersome with traditional modelling methods, the block modelling method can handle arc models for any size of networks. The main advantage of applying the block modelling method is that the computation of the analytical Jacobian matrix is possible and cheap for any number of arc models. The computation time is smaller with this approach than with the traditional approach.*

**Keywords:** Arc models, transients, circuit breakers.

## I. INTRODUCTION

USUALLY current interruption in power systems is simulated by non-ideal switching devices, but these models are too simple to include thermal re-ignition of circuit breakers. Arc models have been developed and they can be used instead of non-ideal switching devices. The simulation with arc models in power systems for current interruption is done under specific conditions, for a small power system and with only a single arc model. The scope of this paper is to show that the simulation of more arc models in large power systems can be performed with the help of the block modelling method.

In literature, several arc models are described[1]. In this paper, we consider the two basic arc models, the Cassie model[2] and the Mayr model[3]. The third arc model studied in this paper is the Habedank model[4] which is a combination of Cassie and Mayr model. Arc models are non-linear and as a result, the Jacobian matrix of the system must be computed. Furthermore, time constants of arc models are relatively small and as a consequence, arc models affect the stiffness and smaller time steps need to be used.

Arc models are embedded in several computer programs each using different modelling methods. The nodal analysis method is mostly used for the simulation of large power system which includes arc models[6] and requires

small time steps. The modified nodal analysis is more suitable as in X-trans[7]. Another approach is the cut set method in MatLab/SimPowerSystem that has a library with arc models[8]. However, these options are rather slow for large power systems in particular when more than one arc model is used.

The block modelling method[9], which gives the space state representation, can be useful for the simulation of arc models in power systems. Firstly, the change of conductivity of the arc model is taken instead of using a controlled current source like in [7], [8]. Secondly, it is possible to compute the analytical Jacobian matrix without noticeable effort and a similar approach can be used for most of the arc models in literature [1].

The paper compares power systems of different size and number of arc models. Sample circuits are used with three types of arc models.

The paper is divided into four parts. The first part presents the method to include an arc block model in the space state representation of the block modelling method. The second part describes the process how to compute the analytic Jacobian matrix when arc models are used. The third part presents the different networks with respect to the computation time and in the fourth part, the conclusions are given.

## II. BLOCK MODELLING METHOD WITH ARC MODELS

### A. Arc models

Cassie and Mayr developed their equations based on the thermal processes taking place inside an electrical arc. Cassie model[2] and Mayr model[3] describe the different parts of an electrical arc, the steady state voltage and the current interruption by gradually changing the conductivity of the arc model ( $g_c$  for the Cassie model and  $g_m$  for the Mayr model). The change of conductivity described by Cassie model is performed by the following differential equation:

$$\frac{dg_c}{dt} = \frac{g_c}{\tau_c} \left( \frac{u_{arc}^2}{U_c^2} - 1 \right) \quad (1)$$

while the Mayr equation is described by the following equation:

$$\frac{dg_m}{dt} = \frac{g_m}{\tau_m} \left( \frac{u_{arc} i_{arc}}{P_m} - 1 \right) \quad (2)$$

where  $i_{arc}$  is the current through the arc,  $u_{arc}$  voltage across the arc and,  $\tau_c$  and  $\tau_m$  are the time constant of the Cassie and Mayr model. Furthermore,  $U_c$  is the steady state

---

R. Thomas and L. van der Sluis are with the Intelligent Electrical Power Grids department, TU DELFT, the Netherlands (r.thomas@tudelft.nl, l.vanderSluis@tudelft.nl).

D. Lahaye and C. Vuik are with the Numerical Analysis department, TU DELFT, the Netherlands (d.j.p.lahaye@tudelft.nl, c.vuik@tudelft.nl)

Paper submitted to the International Conference on Power Systems Transients (IPST2015) in Cavtat, Croatia June 15-18, 2015

voltage and  $P_m$  is the steady state power loss.

The series connection of the Cassie and Mayr model is considered as the Habedank model[4]. As the result, the resistivity of the electrical arc in this model is given by

$$r_{arc} = \frac{1}{g_{arc}} = \frac{1}{\frac{1}{g_c} + \frac{1}{g_m}} \quad (3)$$

and  $i_{arc}$  is defined as:

$$i_{arc} = \frac{u_{arc}}{r_{arc}} = g_{arc}u_{arc} \quad (4)$$

### B. Mathematical expression

The block modelling method[9] gives the space state representation:

$$\dot{x} = f(x, t) = Ax + Be(t) \quad (5)$$

By introducing arc models, the previous equation becomes non-linear and (5) is redefined as:

$$\begin{aligned} \dot{x} &= f(x, t) = Ax + A_{non}(x) + Bg(t) + v(x) \\ &= (\hat{A} + \tilde{A})x + A_{non}(x) + (\hat{B} + \tilde{B})g(t) \\ &\quad + v(x) \end{aligned} \quad (6)$$

where

- $p_1 \in \mathbb{N}$  is the number of inductances;
- $p_2 \in \mathbb{N}$  is the number of capacitances;
- $p_3 \in \mathbb{N}$  is the number of non-linear conductivities due to arc models;
- $p \in \mathbb{N}$  is the number of differential variables ( $p_1 + p_2 + p_3$ );
- $s \in \mathbb{N}$  is the number of sources;
- $x \in \mathbb{R}^p$  is the state vector;
- $e(t) \in \mathbb{R}^s$  is the time dependent input vector;
- $A \in \mathbb{R}^{p \times p}$  is the state matrix;
- $\hat{A} \in \mathbb{R}^{p \times p}$  is the block state matrix;
- $\tilde{A} \in \mathbb{R}^{p \times p}$  is the connection state matrix;
- $B \in \mathbb{R}^{p \times s}$  is the input matrix;
- $\hat{B} \in \mathbb{R}^{p \times s}$  is the block input matrix;
- $\tilde{B} \in \mathbb{R}^{p \times s}$  is the connection input matrix;
- $A_{non}(x) \in \mathbb{R}^{p \times p}$  is the non-linear state matrix;
- $v(x) \in \mathbb{R}^p$  is the non-linear vector.

For the block modelling method, it is necessary to develop the matrix expression of matrices  $\hat{A}_i$  and  $\hat{B}_i$  and the vector source  $e_i$  of each element of the considered power system. As a consequence, the matrices  $\hat{A}_{arc} \in \mathbb{R}^{l \times l} = 0$ ,  $\hat{B}_{arc} \in \mathbb{R}^{l \times l} = 0$  and the vector  $e_{arc} \in \mathbb{R}^1 = 0$  for each arc model of the power system and where  $l$  represents the number of differential variables of the particular arc model. In fact, from (6), only the matrix  $A_{non}$  and the vector  $v$  need to be updated at each time step when arc models are present and actives.

The assumption is that arc models can only be placed between two block models or between a terminal of a block model and the ground. The delta or star connection [9] between block models by the aim of arc models can be realized. However, this requires more calculation time and is more complex.

As for the block modelling method[9], mapping functions are necessary. In fact, for updating the system of

equations, three mapping functions ( $I_{arc}(n, m)$ ,  $C_{arc}(n, m)$  and  $Arc(n, q)$ ) are used. Firstly, let us consider a network composed of  $NBarc$  arc models, each arc model is composed of  $NBmax$  differential variables and is connected to  $NBm$  block models. The first mapping function ( $I_{arc}(n, m)$ ) links the arc model number  $n$  ( $1 \leq n \leq NBarc$ ) and the block number  $m$  ( $1 \leq m \leq NBm$ ) to the position of the state variable of the link capacitance of the block model  $m$ . Although the second function ( $C_{arc}(n, m)$ ) has the same input as the first function, the second function gives the value of the link capacitance of the block model  $m$ . The variable  $NBm$  is all the time equal to two according to our assumption. In the case of an arc model block placed between a block model and the ground, the value  $I_{arc}(n, 2)$  and  $C_{arc}(n, 2)$  does not exist as well as all values associate to them. The last mapping function ( $Arc(n, q)$ ) associates the arc model number  $n$  ( $1 \leq n \leq NBarc$ ) and the differential variable  $q$  ( $1 \leq q \leq NBmax$ ) to its state variable position. Another important information is given by the vector  $\alpha_{arc} \in \mathbb{N}^{NBarc}$  which contains the information whether the  $n^{th}$  arc model is active (1) or inactive (0). Finally,  $\tau_c(n)$ ,  $U_c(n)$ ,  $\tau_m(n)$  and  $P_m(n)$  correspond with the  $n^{th}$  arc model parameters of the considered network.

### C. Formulations

This paragraph focuses on the formulation of (6) when an arc block model is used in a block diagram. For example, let us consider the following sample block diagram composed of a generator, a load and an arc model which is placed between the generator and the load (Fig. 1).



Fig. 1 Simple block diagram with an arc block

The block generator model[9] has as parameters  $R_1$ ,  $R_c$ ,  $L_1$ ,  $C_1$  and  $e(t)$ . The block load model[9] has as parameters  $R_2$ ,  $L_2$ ,  $C_2$  and  $R_d$  which is in parallel with  $C_2$ . The resistance  $R_d$  is added for damping the voltage  $v_{c_2}$ . The arc model is active only when  $\alpha_{arc}(1) = 1$ . The arc model used in a first time is the Cassie model and it has as parameters  $t_c$  and  $U_{arc}$ . The space state representation of Figure (1) is non-linear and can be expressed as:

$$\dot{x} = \begin{bmatrix} -\frac{R_1}{L_1} & -\frac{1}{L_1} & 0 & 0 & 0 \\ \frac{1}{C_1} & -\frac{1}{R_c C_1} & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{R_d C_2} & -\frac{1}{C_2} & 0 \\ 0 & 0 & -\frac{1}{L_2} & -\frac{R_2}{L_2} & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} i_{L_1} \\ v_{C_1} \\ v_{C_2} \\ i_{L_2} \\ g_c \end{bmatrix}$$

$$\begin{aligned}
& + \begin{bmatrix} 1 \\ L \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} e(t) + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{g_c}{C_1} & \frac{g_c}{C_1} & 0 & 0 \\ 0 & \frac{g_c}{C_1} & -\frac{g_c}{C_1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} i_{L_1} \\ v_{C_1} \\ v_{C_2} \\ i_{L_2} \\ g_c \end{bmatrix} \\
& + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \alpha_{arc} \frac{g_c}{\tau_c} \left( \frac{(v_{arc})^2}{U_c^2} - 1 \right) \end{bmatrix} \quad (7)
\end{aligned}$$

while the Cassie model is replaced by the Mayr model with its arc parameters  $\tau_m$  and  $P_m$ , the new system of equations becomes:

$$\begin{aligned}
\dot{x} = & \begin{bmatrix} -\frac{R_1}{L_1} & -\frac{1}{L_1} & 0 & 0 & 0 & 0 \\ \frac{1}{C_1} & -\frac{1}{R_c C_1} & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{R_d C_2} & -\frac{1}{C_2} & 0 & 0 \\ 0 & 0 & \frac{1}{L_2} & -\frac{R_2}{L_2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} i_{L_1} \\ v_{C_1} \\ v_{C_2} \\ i_{L_2} \\ g_c \end{bmatrix} \\
& + \begin{bmatrix} 1 \\ L \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} e(t) + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{g_m}{C_1} & \frac{g_m}{C_1} & 0 & 0 \\ 0 & \frac{g_m}{C_2} & -\frac{g_m}{C_2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} i_{L_1} \\ v_{C_1} \\ v_{C_2} \\ i_{L_2} \\ g_c \end{bmatrix} \\
& + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \alpha_{arc} \frac{g_m}{\tau_m} \left( \frac{u_{arc}^2}{P_m g_m} - 1 \right) \end{bmatrix} \quad (8)
\end{aligned}$$

Finally, the Mayr model is substituted by the Habedank model. The parameters of the Habedank model are  $t_c$ ,  $U_{arc}$ ,  $\tau_m$  and  $P_m$ . The system of equations becomes:

$$\begin{aligned}
\dot{x} = & \begin{bmatrix} -\frac{R_1}{L_1} & -\frac{1}{L_1} & 0 & 0 & 0 & 0 \\ \frac{1}{C_1} & -\frac{1}{R_c C_1} & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{R_d C_2} & -\frac{1}{C_2} & 0 & 0 \\ 0 & 0 & \frac{1}{L_2} & -\frac{R_2}{L_2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} i_{L_1} \\ v_{C_1} \\ v_{C_2} \\ i_{L_2} \\ g_c \\ g_m \end{bmatrix} \\
& + \begin{bmatrix} 1 \\ L \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} e(t) + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{g_{arc}}{C_1} & \frac{g_{arc}}{C_1} & 0 & 0 & 0 \\ 0 & \frac{g_{arc}}{C_2} & -\frac{g_{arc}}{C_2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} i_{L_1} \\ v_{C_1} \\ v_{C_2} \\ i_{L_2} \\ g_c \\ g_m \end{bmatrix}
\end{aligned}$$

$$+ \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \frac{\alpha_{arc}}{\tau_c} \left( \frac{(g_{arc} v_{arc})^2}{U_c^2 g_c} - g_c \right) \\ \frac{\alpha_{arc}}{\tau_m} \left( \frac{(g_{arc} v_{arc})^2}{P_m} - g_m \right) \end{bmatrix} \quad (9)$$

where  $g_{arc} = \frac{g_c g_m}{g_c + g_m}$

In this example, matrices  $\tilde{A}$  and  $\tilde{B}$  do not exist due to block models used and to the block diagram on Fig. 1.

#### D. Block modelling method algorithm

Let us use the example of Fig. 1 to complete the algorithm. Firstly, let us apply the block modelling method definition [8] for computing matrices  $A$  and  $B$ .

$$A = \hat{A} = \begin{bmatrix} \hat{A}_G & 0 & 0 \\ 0 & \hat{A}_L & 0 \\ 0 & 0 & \hat{A}_{arc} \end{bmatrix} \quad (10)$$

$$B = \hat{B} = \begin{bmatrix} \hat{B}_G & 0 & 0 \\ 0 & \hat{B}_L & 0 \\ 0 & 0 & \hat{B}_{arc} \end{bmatrix} \quad (11)$$

Now, we can develop the different mapping function of Fig. 1 according to the definition stated in part II. B. Figure 1 has one arc block model ( $NBarc = 1$ ) which connects the two block models ( $m = 2$ ). As consequence, we can write  $I_{arc}(1,1) = 2$ ,  $I_{arc}(1,2) = 3$ ,  $C_{arc}(1,1) = C_1$ ,  $C_{arc}(1,2) = C_2$ . The Cassie model and the Mayr model have one differential equation so ( $q = 1$ ) so  $Arc(1,1) = 5$ . In the case of the Habedank model,  $q = 2$  so the  $Arc$  mapping function is  $Arc(1,1) = 5$  and  $Arc(1,2) = 6$ .

At the start of the simulation, the matrix  $A_{non}(x)$  and the vector  $v$  are initialized to zero. Before updated the previous matrix and vector, if we consider  $NBarc$  models ( $1 \leq n \leq NBarc$ ) in the considered power system, we can write the conductivity of each arc model according to their type such as:

- $g_c(n) = x(Arc(n, 1))$ , in the case of the Cassie model;
- $g_m(n) = x(Arc(n, 1))$ , in the case of the Mayr model;
- $g_h(n) = \frac{g_{hc}(n)g_{hm}(n)}{g_{hc}(n)+g_{hm}(n)}$  where  $g_{hc}(n) = x(Arc(n, 1))$  and  $g_{hm}(n) = x(Arc(n, 2))$  in the case of the Habedank model.

Another important parameter to compute, before updating the matrix  $A_{non}$  and the vector  $v$ , is the voltage across the arc block models and can be written as:

$$u_{arc}(n) = x(I_{arc}(n, 1)) - x(I_{arc}(n, 2)) \quad (12)$$

In order to update the matrix  $A_{non}(x)$ , the following expressions are used for any of the previous arc model for any number of arc models for  $1 \leq n \leq NBarc$ .

$$A_{non}(I_{arc}(n, 1), I_{arc}(n, 1)) = -\frac{g_{arc}}{C_{arc}(1,1)} \quad (13)$$

$$A_{non}(I_{arc}(n, 1), I_{arc}(n, 2)) = \frac{g_{arc}}{C_{arc}(1,1)} \quad (14)$$

$$A_{non}(I_{arc}(n, 2), I_{arc}(n, 2)) = -\frac{g_{arc}}{C_{arc}(1,2)} \quad (15)$$

$$A_{non}(I_{arc}(n, 2), I_{arc}(n, 1)) = \frac{g_{arc}}{C_{arc}(1,2)} \quad (16)$$

where  $g_{arc} = g_c(n)$  or  $g_m(n)$  or  $g_h(n)$  according to the type of the  $n^{th}$  arc model.

In the case of the Cassie model the element associate to it and to the vector  $v$  is:

$$v(I_{pos}(n, 1)) = \alpha_{arc}(n) \frac{g_c(n)}{\tau_c} \left( \frac{u_{arc}(n)^2}{U_c(n)^2} - 1 \right) \quad (17)$$

while for the Mayr model

$$v(I_{pos}(n, 1)) = \alpha_{arc}(n) \frac{g_m(n)}{\tau_m(n)} \left( \frac{u_{arc}(n)^2}{P_m(n)g_m(n)} - 1 \right) \quad (18)$$

finally, the expressions if the Habedank model is used are:

$$v(I_{pos}(n, 1)) = \frac{\alpha_{arc}(n)}{\tau_c(n)} \left( \frac{(g_h(n)u_{arc}(n))^2}{U_c(n)^2 g_{hc}(n)} - g_{hc}(n) \right) \quad (19)$$

$$v(I_{pos}(n, 2)) = \frac{\alpha_{arc}(n)}{\tau_m(n)} \left( \frac{(g_h(n)u_{arc}(n))^2}{P_m(n)} - g_{hm}(n) \right) \quad (20)$$

### III. JACOBIAN MATRIX

The Jacobian matrix  $J$  is computed as:

$$J = \begin{bmatrix} \frac{\partial f_1(x, t)}{\partial x_1} & \dots & \frac{\partial f_1(x, t)}{\partial x_p} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_p(x, t)}{\partial x_1} & \dots & \frac{\partial f_p(x, t)}{\partial x_p} \end{bmatrix} \quad (21)$$

$$\text{where } f(x, t) = \begin{bmatrix} f_1(x, t) \\ \vdots \\ f_p(x, t) \end{bmatrix}$$

There are two ways to compute the Jacobian matrix, with the numerical method[10] or with the analytical method. In some cases, it is not possible to compute analytically the Jacobian matrix and so the numerical method is used. However, when it is possible to compute the Jacobian matrix analytically, this is usually faster and cheaper than the numerical approach. In our case, the Jacobian matrix can be found analytically. Furthermore, the same information as for upgrading the matrix  $A_{non}$  and the vector  $v$  is used.

According to our formulation (6) and the Jacobian matrix formulation (21), the Jacobian matrix considered is defined as:

$$J = A + \begin{bmatrix} \frac{\partial \sum_{j=1}^p A_{non}(1, j)x_j}{\partial x_1} & \dots & \frac{\partial \sum_{j=1}^p A_{non}(1, j)x_j}{\partial x_p} \\ \vdots & \ddots & \vdots \\ \frac{\partial \sum_{j=1}^p A_{non}(1, j)x_j}{\partial x_1} & \dots & \frac{\partial \sum_{j=1}^p A_{non}(1, j)x_j}{\partial x_p} \end{bmatrix}$$

$$+ \begin{bmatrix} \frac{\partial v(1)}{\partial x_1} & \dots & \frac{\partial v(p)}{\partial x_p} \\ \vdots & \ddots & \vdots \\ \frac{\partial v(p)}{\partial x_1} & \dots & \frac{\partial v(p)}{\partial x_p} \end{bmatrix} = A + J_{non} + J_v \quad (22)$$

The computation of the matrices  $J_{non}$  and  $J_v$  can be done by the following process (23)-(46) for  $1 \leq n \leq NBarc$  according to the type of  $n^{th}$  arc model used. The first step is to initialize the matrices  $J_{non}$  and  $J_v$  to zero.

$$J_{non}(I_{arc}(n, 1), I_{arc}(n, 1)) = -\frac{g_{arc}}{C_{arc}(n, 1)} \quad (23)$$

$$J_{non}(I_{arc}(n, 1), I_{arc}(n, 2)) = \frac{g_{arc}}{C_{arc}(n, 1)} \quad (24)$$

$$J_{non}(I_{arc}(n, 2), I_{arc}(n, 2)) = \frac{g_{arc}}{C_{arc}(n, 2)} \quad (25)$$

$$J_{non}(I_{arc}(n, 2), I_{arc}(n, 1)) = -\frac{g_{arc}}{C_{arc}(n, 2)} \quad (26)$$

where  $g_{arc} = g_c(n)$  or  $g_m(n)$  or  $g_h(n)$  according to the type of arc model used. For the Cassie model and Mayr model, the two following equations are used for upgrading the matrix  $J_{non}$ .

$$J_{non}(I_{arc}(n, 1), I_{pos}(n, 1)) = -\alpha_{arc}(n) \frac{u_{arc}(n)}{C_{arc}(1,1)} \quad (27)$$

$$J_{non}(I_{arc}(n, 2), I_{pos}(n, 1)) = \alpha_{arc}(n) \frac{u_{arc}(n)}{C_{arc}(1,2)} \quad (28)$$

while for the Habedank model, the four following equations need to be computed and upgraded.

$$J_{non}(I_{arc}(n, 1), I_{pos}(n, 1)) = -\frac{\alpha_{arc}(n)u_{arc}(n)g_h(n)^2}{C_{arc}(1,1)g_{hc}(n)^2} \quad (29)$$

$$J_{non}(I_{arc}(n, 1), I_{pos}(n, 2)) = -\frac{\alpha_{arc}(n)u_{arc}(n)g_h(n)^2}{C_{arc}(1,1)g_{hm}(n)^2} \quad (30)$$

$$J_{non}(I_{arc}(n, 2), I_{pos}(n, 1)) = \frac{\alpha_{arc}(n)u_{arc}(n)g_h(n)^2}{C_{arc}(1,2)g_{hc}(n)^2} \quad (31)$$

$$J_{non}(I_{arc}(n, 2), I_{pos}(n, 2)) = \frac{\alpha_{arc}(n)u_{arc}g_h(n)^2}{C_{arc}(1,2)g_{hm}(n)^2} \quad (32)$$

The last step of the computation of the Jacobian matrix  $J$  is the computation of the Jacobian of the vector  $v$  called matrix  $J_v$ . For the Cassie model, the following three equations are considered for upgrading the matrix  $J_v$ .

$$J_v(I_{pos}(n, 1), I_{arc}(n, 1)) = \alpha_{arc}(n) \frac{2g_c(n)u_{arc}(n)}{\tau_c(n)U_c(n)^2} \quad (33)$$

$$J_v(I_{pos}(n, 1), I_{arc}(n, 2)) = -\alpha_{arc}(n) \frac{2g_c(n)u_{arc}(n)}{\tau_c(n)U_c^2(n)} \quad (34)$$

$$J_v(I_{pos}(n, 1), I_{pos}(n, 1)) = \frac{\alpha_{arc}(n)}{\tau_c(n)} \left( \frac{u_{arc}^2(n)}{U_c(n)^2} - 1 \right) \quad (35)$$

While for Mayr, the following three equations are considered.

$$J_v(I_{pos}(n, 1), I_{arc}(n, 1)) = \alpha_{arc}(n) \frac{2g_m(n)^2 u_{arc}(n)}{\tau_m(n) P_m(n)} \quad (36)$$

$$J_v(I_{pos}(n, 1), I_{arc}(n, 2)) = -\alpha_{arc}(n) \frac{2g_m(n)^2 u_{arc}(n)}{\tau_m(n) P_m(n)} \quad (37)$$

$$J_v(I_{pos}(n, 1), I_{pos}(n, 1)) = \frac{\alpha_{arc}(n)}{\tau_m(n)} \left( \frac{2g_m(n) u_{arc}^2(n)}{P_m(n)} - 1 \right) \quad (38)$$

Finally, for the Habedank model, the following eight equations are used.

$$J_v(I_{pos}(n, 1), I_{arc}(n, 1)) = \alpha_{arc}(n) \frac{2g_h(n)^2 u_{arc}(n)}{\tau_c(n) U_c^2(n) g_{hc}(n)} \quad (39)$$

$$J_v(I_{pos}(n, 1), I_{arc}(n, 2)) = -\alpha_{arc}(n) \frac{2g_h(n)^2 u_{arc}(n)}{\tau_c(n) U_c(n)^2 g_{hm}(n)} \quad (40)$$

$$J_v(I_{pos}(n, 1), I_{pos}(n, 1)) = \frac{\alpha_{arc}(n)}{\tau_c(n)} \left( \frac{u_{arc}(n)^2 g_h(n)}{U_c(n)^2 g_{hc}(n)^2} \left( 1 - \frac{2g_h(n)}{g_{hm}(n)} \right) - 1 \right) \quad (41)$$

$$J_v(I_{pos}(n, 1), I_{pos}(n, 2)) = \frac{\alpha_{arc}(n) 2u_{arc}(n)^2 g_h(n)^3}{\tau_c(n) U_c(n)^2 g_{hc}(n) g_{hm}(n)^2} \quad (42)$$

$$J_v(I_{pos}(n, 2), I_{arc}(n, 1)) = \alpha_{arc}(n) \frac{2g_h(n)^2 u_{arc}(n)}{\tau_m(n) P_m(n)} \quad (43)$$

$$J_v(I_{pos}(n, 2), I_{arc}(n, 2)) = -\alpha_{arc}(n) \frac{2g_h(n)^2 u_{arc}(n)}{\tau_m(n) P_m(n)} \quad (44)$$

$$J_v(I_{pos}(n, 2), I_{pos}(n, 1)) = \frac{\alpha_{arc}(n) 2u_{arc}(n)^2 g_h(n)^3}{\tau_m(n) P_m(n) g_{hc}(n)^2} \quad (45)$$

$$J_v(I_{pos}(n, 2), I_{pos}(n, 2)) = -\frac{\alpha_{arc}(n)}{\tau_m(n)} \left( \frac{2u_{arc}(n)^2 g_h(n)^3}{P_m(n) g_{hm}(n)^2} - 1 \right) \quad (46)$$

#### IV. TEST CASES

##### A. Simple Circuit

Let us consider again Fig. 1 modelled with the block modelling method implemented in MatLab for the different arc models and modelled in MatLab/SimPowerSystem. After that, the computation time of both modelling methods are compared. In fact, the integration method is the same for both modelling methods and this integration method is called *ode23tb*[11] with a relative tolerance of  $10^{-4}$ .

The parameters of the different simple block diagram according to Equation to are  $R_1 = R_2 = 0.01\Omega$ ,  $R_c = R_d = 1k\Omega$ ,  $L_1 = L_2 = 3.52mH$ ,  $C_1 = 19.8nF$ ,  $C_2 = 19.8\mu F$ ,  $e(t) = 59196 \cos(100\pi t) V$ ,  $t_c = 12\mu s$ ,  $U_{arc} = 5kV$ ,  $\tau_m = 4\mu s$  and  $P_m = 2MW$ . The arc model is active when the time exceeds of 0.012s.

Table 1 shows that the computation time of the sample electrical block diagram of Fig. 1 is the shortest when the Jacobian matrix is computed analytically. In fact, it is in general cheaper to use an analytical Jacobian instead of a numerical Jacobian. Fig. 2 shows the current through the Cassie model and the voltage across it. As described in the literature, the Cassie model maintains the steady state voltage. Fig. 3 shows the current through the Mayr model and the voltage across it. As described in the literature, the Mayr model is able to interrupt short-circuit current. Fig. 4 shows the current through the Habedank model and the voltage across it. The arc parameters used permit the interruption of current.

Table 1  
Computation time of the block diagram of Fig. 1 in second for different modelling methods and arc models

	Cassie model	Mayr model	Habedank model
MatLab/SimPower Numerical Jacobian	2.61 s	2.53 s	2.96 s
Block modelling Numerical Jacobian	1.30 s	1.99 s	2.05 s
Block modelling Analytical Jacobian	1.12 s	1.92 s	1.99 s

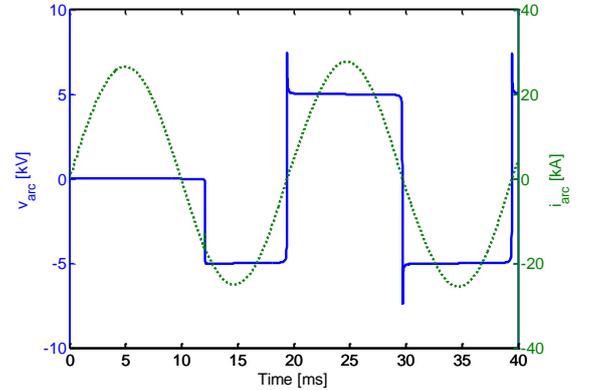


Fig. 2 Voltage and Current of the Cassie model

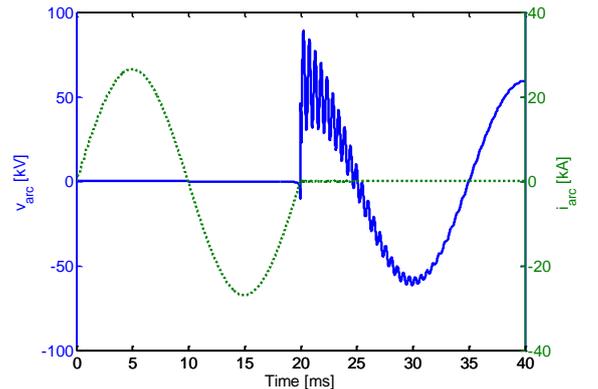


Fig. 3 Voltage and Current of the Mayr model

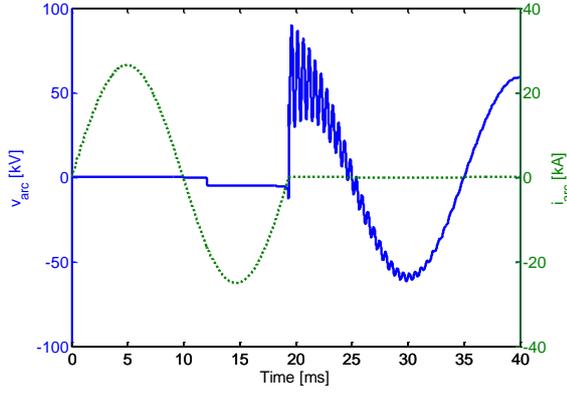


Fig. 4 Voltage and Current of the Habedank model

### B. Generator fault

For the Generator fault, we consider the following block diagram.



Fig. 5 Sample block diagram

The block diagram of Fig. 5 simulates the interruption of a three phases to ground short-circuit (modelled by three arc models according to the symbol ARC 3p of Fig. 5) by the three phases generator (symbol G 3p in Fig. 5 which consist of three independent generator blocks[8]). Each phase of the generator block model has four parameters  $R = 0.01\Omega$ ,  $R_c = 1k\Omega$ ,  $L = 3.52mH$  and  $C = 19.8nF$ . We consider the voltage supplied by the voltage source of each phase to be shifted of  $\pm 120^\circ$  between each other:  $e_1(t) = 59196 \cos(100\pi t) V$ ,  $e_2(t) = 59196 \cos\left(100\pi t - \frac{2\pi}{3}\right) V$  and  $e_3(t) = 59196 \cos\left(100\pi t + \frac{2\pi}{3}\right) V$ . Arc model parameters are the same for the three arc models and they are  $t_c = 12\mu s$ ,  $U_{arc} = 5kV$ ,  $\tau_m = 4\mu s$  and  $P_m = 2MW$ . They are active when the simulation time will be larger than 0.012s. As in the previous test case, we compare the different computation times for different modelling methods and arc models.

From Table 2, we can conclude that the analytical Jacobian is more efficient than the numerical Jacobian. Moreover, the MatLab/SimPowerSystem modelling method is not the best method especially when the number of arc models increase.

Table 2

Computation time of the block diagram of Fig. 5 in second for different modelling methods and arc models

	Cassie model	Mayr model	Habedank model
MatLab/SimPower Numerical Jacobian	5.71 s	3.35 s	3.84 s
Block modelling Numerical Jacobian	1.03 s	0.82 s	0.82 s
Block modelling Analytical Jacobian	0.91 s	0.73 s	0.76 s

### C. Larger test cases

In this part, we consider the following parameters for each block model type [9] of the block diagram of Fig. 6:

- Generator:  $R = 0.001\Omega$ ,  $R_c = 100\Omega$ ,  $L = 3.52mH$  and  $C = 1.98\mu F$ ,  

$$e(t) = \begin{cases} 10t * 59196 \cos(100\pi t) & \text{if } t < 0.1 \\ 59196 \cos(100\pi t) & \text{else} \end{cases}$$
- Load:  $R = 1\Omega$ ,  $R_d = 100\Omega$ ,  $L = 35.2mH$  and  $C = 1.98\mu F$
- PI-section:  $R = 10\mu\Omega$ ,  $R_d = 100\Omega$ ,  $L = 20\mu H$  and  $C_1 = C_2 = 20\mu F$ ;
- PII-section:  $R_1 = R_2 = 10\mu\Omega$ ,  $R_d = 100\Omega$ ,  $L_1 = L_2 = 20\mu H$  and  $C_1 = C_2 = C_3 = 20\mu F$ ;
- Arc:  $t_c = 12\mu s$ ,  $U_{arc} = 5kV$ ,  $\tau_m = 4\mu s$  and  $P_m = 2MW$ .

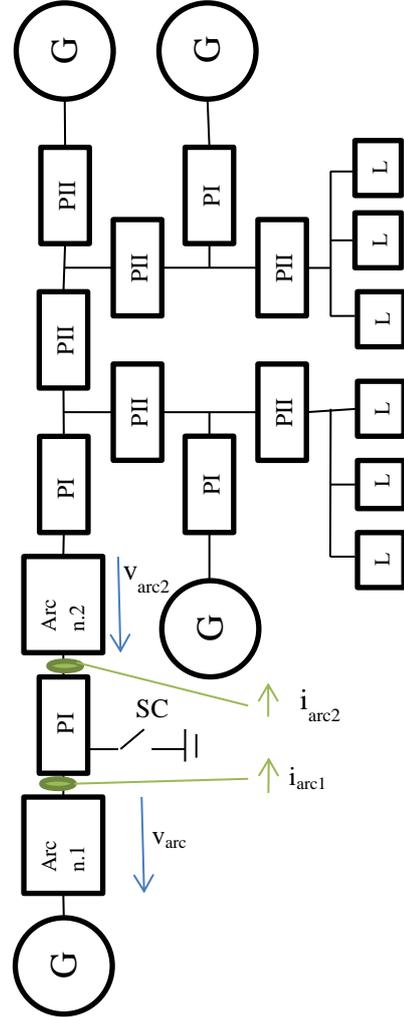


Fig. 6 Electrical block diagram network 1

Each switching device that links block models has an internal resistance of  $0.1m\Omega$ . All  $R_d$  resistances are in parallel with link capacitors of each block model terminal of the Load, PI-section and PII-section block models.

Let us first consider the block model diagram of Fig. 6. Moreover, arc models used are Habedank models. In fact, the block model diagram is composed of 66 differential variables ( $x \in \mathbb{R}^{66}$ ). The scenario of this test case is:

- From  $t=0s$  to  $t<0.345s$  : All currents and voltages of the block model diagram of Figure 6 reach their steady state;
- At  $t=0.345s$ : A short circuit appears by closing the switching devise SC;
- From  $t>0.245s$  to  $t<0.5s$ : The short circuit current appears and all currents and voltages want to go to the new steady state;
- At  $t=0.5s$ : Both arc models are activated;
- From  $t>0.5s$  to  $t=1s$ : Arc models do interrupt or do not the short-circuit current according to the parameters of the block diagram

For the block diagram of Fig. 6, we are going to compare the modelling method with MatLab and integrating the system of differential equations with the method called *ode23tb* with a relative tolerance of  $10^{-3}$  with the numerical and with the analytical Jacobian.

From Table 3, the computation is four times faster with the analytical Jacobian than with the numerical Jacobian. From Fig. 7 to 9, we can see the development of the scenario with the interruption of current of the both sides of the PI-section which contains the short circuit. In Fig. 9, we can see that, once arc n.2 interrupts the short-circuit current, the voltage and the current of the load reach the new steady state because the topology of the power system has changed.

Table 3  
Computation time

Numerical Jacobian in MatLab	Analytical Jacobian in MatLab
90.3s	20.3s

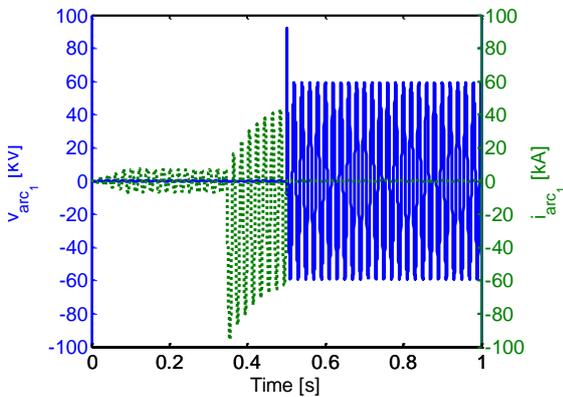


Fig. 7 Current and voltage of the arc n.1

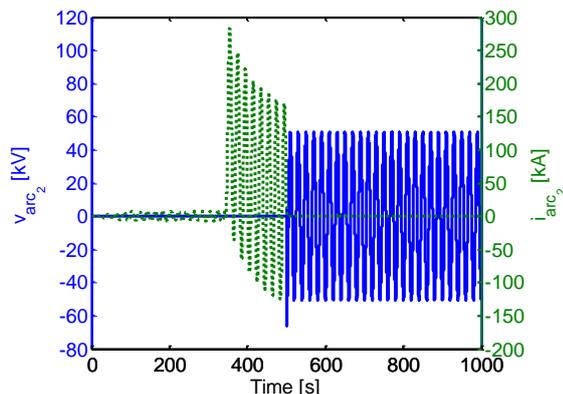


Fig. 8 Current and voltage of the arc n.2

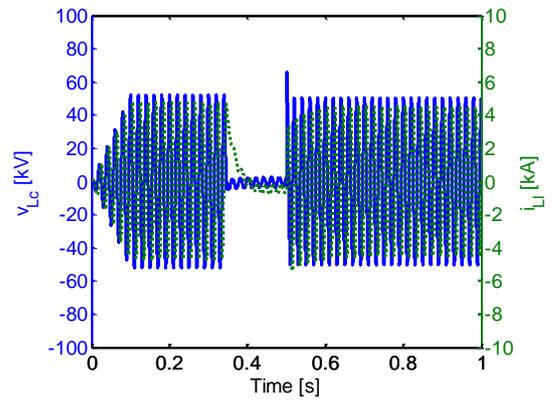


Fig. 9 Current and voltage of one load

When we use a second time the same block diagram and numerical integration methods. But, this time we change the generator parameter values of  $C$  and  $R_c$  by  $C = 1.98nF$  and  $R_c = 1k\Omega$ .

Table 4  
Computation time

Numerical Jacobian in MatLab	Analytical Jacobian in MatLab
95.3s	60.3s

By changing these two values, the speed up is only of 1.5 times in MatLab. This is due to the fact that arc n.1 never interrupts the short circuit current supplied by the generator as shown on Fig. 10. But, as shown on Fig. 11, arc n.2 interrupts the short circuit current.

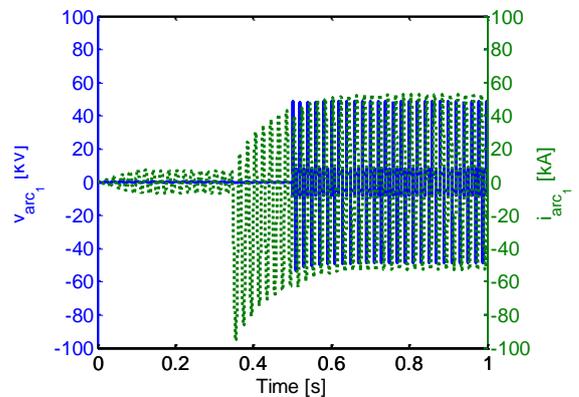


Fig. 10 Current and voltage of the arc n.1

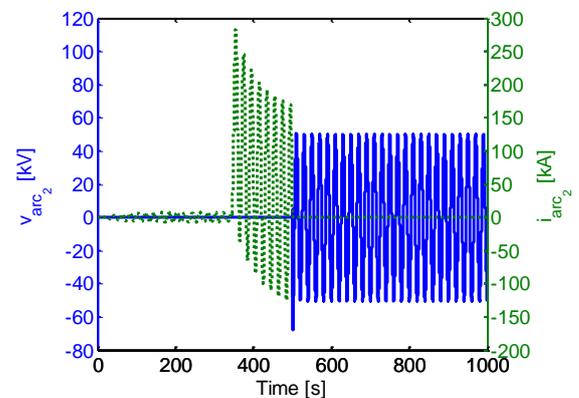


Fig. 11 Current and voltage of the arc n.2

The difference in speed up time between the cases of the block model diagram of Fig. 6 is due to the fact of the non-interruption of the short circuit current by the arc n.1.

