

Principes de la simulation électrique

Table des matières

1	Introduction	3
2	Algorithmique	4
2.1	Formulation du problème	4
2.2	Les trois analyses basiques de la simulation électrique	5
2.2.1	L'analyse statique	5
2.2.2	L'analyse transitoire	5
2.2.3	L'analyse alternative petit signal	5
2.3	Représentation fonctionnelle d'un simulateur électrique	6
2.4	Erreurs associées	6
2.4.1	Erreur de modélisation	6
2.4.2	Erreur de méthode	7
2.4.3	Erreur de troncature	7
2.5	Le conditionnement des méthodes	8
2.6	Les limites d'utilisation d'un simulateur	8
3	Méthodes de formulation des équations	8
3.1	Les lois de Kirchhoff	8
3.2	L'équation de constitution généralisée	9
3.3	L'analyse nodale	10
3.3.1	Les différentes méthodes de formulation des équations	10
3.3.2	La matrice admittance indéfinie	11
3.3.3	La matrice admittance définie	13
3.3.4	Le traitement des sources contrôlées	14
3.3.5	Un exemple de "manipulation" matricielle	14
3.4	L'analyse nodale modifiée (la MNA)	16
4	Méthodes de résolution d'un système d'équations linéaires	17
4.1	L'algorithme de la remontée	18
4.2	L'algorithme d'élimination de Gauss	18
4.3	La décomposition LU	19
4.4	Le conditionnement des méthodes	20
5	Méthodes numériques d'intégration	20
5.1	Les méthodes d'intégration d'Euler explicite et implicite	20
5.2	La méthode d'intégration trapézoïdale	22
5.3	Les méthodes d'intégration de Gear	23
5.4	La traduction des méthodes d'intégration en termes de pôles	23
5.4.1	Intégration implicite d'Euler	24
5.4.2	Intégration trapézoïdale	24
5.4.3	Intégration Gear2	25
5.5	Conditionnement des méthodes d'intégration	26
5.5.1	Contrôle de la LTE	26

5.5.2	Contrôle du pas d'intégration avec commutation d'algorithme . . .	26
5.5.3	Contrôle du pas d'intégration par comptage du nombre d'itérations	26
5.5.4	Adaptation de Gear2	26
5.6	Choix de la variable d'intégration	27
5.7	Choix d'une méthode d'intégration	29
6	Méthodes numériques d'analyse non linéaires	30
6.1	Le principe de l'algorithme de Newton-Raphson	30
6.2	L'algorithme de Newton-Raphson pour un système d'équations	31
6.3	Circuit équivalent temps discret de Newton-Raphson	32
6.4	Critères de convergence de l'algorithme	33
6.5	Conditionnement de la méthode	34
6.5.1	Conditionnement par limitation simple	35
6.5.2	Conditionnement par le principe du continuum	35
6.5.3	Conditionnement par réduction d'erreur et "bypass"	36
6.5.4	Caractéristiques des simulateurs de troisième génération	37

1 Introduction

La taille, la complexité et le degré de raffinement des modèles des dispositifs actifs utilisés, sont devenus tellement importants que la conception d'un circuit intégré est totalement impossible sans l'assistance d'outils informatiques de simulation. En effet, ce n'est que par une évaluation rapide des performances d'un circuit sans avoir à le fabriquer, que l'on peut obtenir un temps de conception réduit entraînant une limitation des coûts de production.

On distingue essentiellement trois catégories de simulateurs :

La première catégorie est constituée des simulateurs logiques ou événementiels qui ne s'intéressent qu'au changement d'état logique des nœuds d'un circuit. Ils sont uniquement utilisés en électronique numérique pour évaluer fonctionnellement les performances. Sur chaque événement, le simulateur n'ayant à examiner que les éléments de circuit affectés par le changement d'état puis éventuellement à le propager, aucune résolution des lois de Kirchhoff et aucune analyse matricielle n'est requise. Ces outils sont ainsi très rapides et n'ont pratiquement aucune limitation en termes de taille et de complexité du circuit à simuler.

La deuxième catégorie de simulateurs est également limitée à l'analyse des circuits numériques, elle est constituée des simulateurs temporels. Les dispositifs non-linéaires sont remplacés par des modèles linéaires par morceaux ou tabulés (*figure 1*), la quasi-unidirectionnalité et le fort taux d'inactivité caractérisant les "gros" circuits numériques (statistiquement plus de 80% du circuit est inactif sur une phase d'horloge) sont exploités. Ce type de simulateur "plus physique" que les simulateurs logiques permet d'obtenir des précisions de l'ordre de quelques pourcent sur les temps de réponse pour un temps de calcul dix à cent fois plus long.

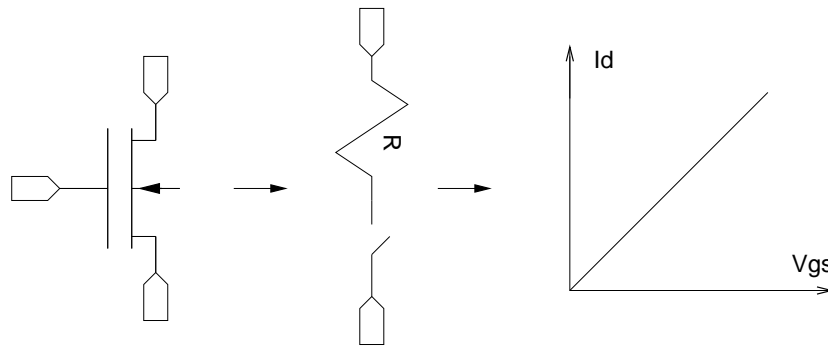


FIGURE 1 –

modélisation linéaire par morceaux du transistor MOS

Quant à la troisième catégorie, elle est constituée des simulateurs analogiques, ou simulateurs électriques, qui ont essentiellement pour vocation l'étude du comportement d'un circuit linéaire ou non-linéaire sur un intervalle de temps continu (ou plus exactement discrétisé avec un pas de calcul théoriquement très inférieur à la plus petite constante de temps du circuit à analyser). Ils sont aussi bien utilisés en électronique numérique pour la caractérisation électrique des bibliothèques de cellules (temps de propagation, consommation, ...), la conception des mémoires et des circuits "rapides", qu'en électronique analogique pour la détermination des caractéristiques fréquentielles et temporelles des circuits. Devant résoudre itérativement les lois de Kirchhoff pour l'intégralité

du circuit et pour tous les composants, c'est un outil lent et limité en terme de taille et de complexité des circuits simulés. Si n est le nombre de dispositifs non-linéaires le temps de calcul est généralement de l'ordre de

$$T_{CPU} \propto n^\alpha \quad \text{avec} \quad 1 < \alpha \leq 2$$

c'est à dire plusieurs milliers de fois moins rapide qu'un simulateur logique et plusieurs centaines de fois moins rapide qu'un simulateur temporel.

La simulation logique et/ou temporelle et/ou analogique peuvent être combinées (dans un même exécutable ou en fonctionnement séparé) pour constituer un type de simulateur appelé simulateurs mixtes, permettant directement l'évaluation des performances électriques des systèmes électroniques mixtes constitués de circuits analogiques et numériques intégrés sur un même substrat, c'est à dire les **SoC** (**S**ysteme **o**n **C**hip).

Enfin, des simulateurs analogiques dédiés ont également été développés pour des applications spécifiques, l'exemple type est le simulateur SWITCAP [?] qui permet l'analyse des circuits à capacités commutées.

Historiquement [?], les premiers simulateurs électriques sont contemporains des premiers ordinateurs puisque c'est dès 1950 qu'un calculateur composé de relais électromécaniques a été programmé pour l'analyse et la conception des filtres électriques. Bien entendu leur évolution a suivi l'évolution des ordinateurs, et nous sommes passés en moins d'un quart de siècle d'un simulateur de première génération permettant difficilement la simulation de circuits composés de quelques transistors modélisés par une dizaine de paramètres à un outil de troisième génération permettant la simulation de circuits de plusieurs milliers de transistors MOS modélisés par une centaine de paramètres [?]. Aujourd'hui, la plupart des simulateurs électriques, en versions commercialisées ou versions publiques, sont des descendants du simulateur de deuxième génération **SPICE**

(**S**imulation **P**rogram with **I**ntegrated **C**ircuit **E**mphasis)) développé et distribué gratuitement par l'université de Berkeley dès 1972. Les simulateurs commerciaux actuellement les plus utilisés sont certainement [?] : HSPICE créé par Meta-Software, ELDO créé par Anacad, SPECTRE créé par Cadence, PSPICE créé par Micro-sim, En termes de simulateurs publiques (et ainsi gratuits) le plus utilisé est sans aucun doute SPICE3F5 distribué par Berkeley (<http://www.eecs.berkeley.edu/>).

2 Algorithmique

2.1 Formulation du problème

Simuler un circuit électronique consiste tout d'abord à lui associer un modèle mathématique puis à calculer numériquement sa réponse à une excitation spécifique. Pour ce faire, on passe du circuit physique à un système d'équations par une *méthode de formulation des équations*, en représentant chaque composant de base (résistance, condensateur, inductance, ...) par son modèle mathématique et en appliquant les contraintes de connexions imposées par les lois de Kirchhoff. Le fonctionnement du circuit peut ainsi être formalisé par un système d'équations différentielles implicite de la forme

$$F(\mathbf{x}, \dot{\mathbf{x}}, t) = 0$$

L'opérateur F est en général un opérateur non linéaire fonction du temps par l'intermédiaire du vecteur $\mathbf{u}(t)$ des sources indépendantes (stimulis), du vecteur \mathbf{x} des variables

inconnues et du vecteur dérivé $\dot{\mathbf{x}}$.

Si cette formulation "standard" est de très loin la plus utilisée actuellement en simulation électrique, ce n'est pas la seule possible. En effet, un circuit constitué de n éléments dynamiques (condensateurs et inductances) peut toujours être mis sous la forme d'une seule équation différentielle d'ordre n pouvant être mise sous une forme canonique dite à variable d'état (ou normale) constituée d'un système explicite des n dérivées du vecteur d'état \mathbf{x} correspondant aux n équations différentielles d'ordre un telles que

$$\dot{\mathbf{x}} = F(\mathbf{x}, t) = [A]\mathbf{x} + \mathbf{b}u,$$

la matrice $[A]$ étant la matrice d'état. Ce second formalisme, qui permet d'utiliser directement l'arsenal mathématique mis au point pour l'étude des systèmes dynamiques [?], est très satisfaisant intellectuellement mais pratiquement, jusqu'à présent il n'a conduit qu'à la réalisation d'outils de simulation pouvant être numériquement instables.

2.2 Les trois analyses basiques de la simulation électrique

Pour caractériser complètement un circuit électrique, au moins trois types d'analyse doivent pouvoir être réalisés par un simulateur électrique : une analyse statique, une analyse transitoire et une analyse alternative petit signal.

2.2.1 L'analyse statique

L'analyse statique est utilisée pour déterminer les points de fonctionnement (la polarisation et les points de repos correspondant) ainsi que les caractéristiques de transfert des circuits électriques linéaires et non-linéaires. Pour ce faire, le vecteur $\dot{\mathbf{x}}$ étant nul, les composants réactifs sont neutralisés (les condensateurs sont remplacés par des circuits ouverts et les inductances sont remplacées par des court-circuits) et toutes les sources indépendantes sont considérées comme statiques. Si l'opérateur F est linéaire la solution est directement déterminée par une *méthode numérique de résolution d'un système d'équations linéaires*, sinon le système est préalablement linéarisé par une *une méthode numérique d'analyse non linéaire*.

2.2.2 L'analyse transitoire

L'analyse transitoire est utilisée pour déterminer la réponse temporelle d'un circuit pour une durée d'observation $[0, T]$. Pour ce faire, l'intervalle $[0, T]$ est discrétisé afin d'obtenir les instants de calcul $[0, t_1, t_2, \dots, T]$. Puis, partant d'un jeu de valeurs initiales spécifié par l'utilisateur ou par une analyse statique, pour chaque instant de calcul une intégration numérique est effectuée par une *méthode numérique d'intégration* afin de transformer le système d'équations différentielles en système d'équations algébriques. Enfin, le système est linéarisé par une *méthode numérique d'analyse non linéaire* pour délivrer la solution par une *méthode numérique de résolution d'un système d'équations linéaires*.

2.2.3 L'analyse alternative petit signal

L'analyse linéaire alternative petit signal est utilisée pour déterminer les caractéristiques fréquentielles d'un circuit. Pour ce faire, tous les éléments actifs non-linéaires du circuit sont modélisés par un circuit équivalent linéaire petit signal autour d'un point de fonctionnement déterminé par une analyse statique. Tous les stimuli sont sinusoïdaux et

de même fréquence mais peuvent avoir des phases relatives différentes. Les impédances ou les admittances sont mises sous leur forme opérationnelle et évaluées sur l'axe imaginaire du plan de la variable complexe P ($p = j\omega$). Ainsi, l'impédance d'un condensateur est considérée sous la forme opérationnelle $Z_C = 1/jC\omega$, l'impédance d'une inductance est considérée sous la forme $Z_L = jL\omega$, un transistor MOS est considéré avec son [schéma équivalent petit signal](#)

(figure 2), Une méthode numérique de résolution d'un système d'équations linéaires travaillant en arithmétique complexe est ensuite utilisée.

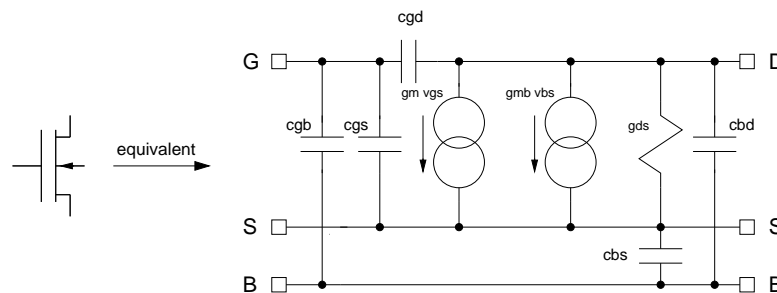


FIGURE 2 –

Schéma équivalent petit signal d'un transistor MOS

2.3 Représentation fonctionnelle d'un simulateur électrique

Un simulateur électrique standard (deuxième génération) est toujours construit à partir d'au moins quatre méthodes numériques

- une méthode de formulation des équations
- une méthode de résolution d'un système d'équations linéaires
- une méthode d'intégration
- une méthode d'analyse non linéaire,

il est représentable par le diagramme fonctionnel de la *figure 3*.

2.4 Erreurs associées

Par principe, tous les résultats de simulation, sont biaisés par trois sources d'erreur inhérentes à toute analyse numérique [?] :

- l'erreur de modélisation
- l'erreur de méthode
- l'erreur de troncature

2.4.1 Erreur de modélisation

Ce type d'erreur, également appelé erreur inhérente, provient du fait qu'il est pratiquement toujours impossible de mettre sous une forme mathématique rigoureuse un processus physique quelconque. La réduction de l'erreur de modélisation est du domaine du modélisateur.

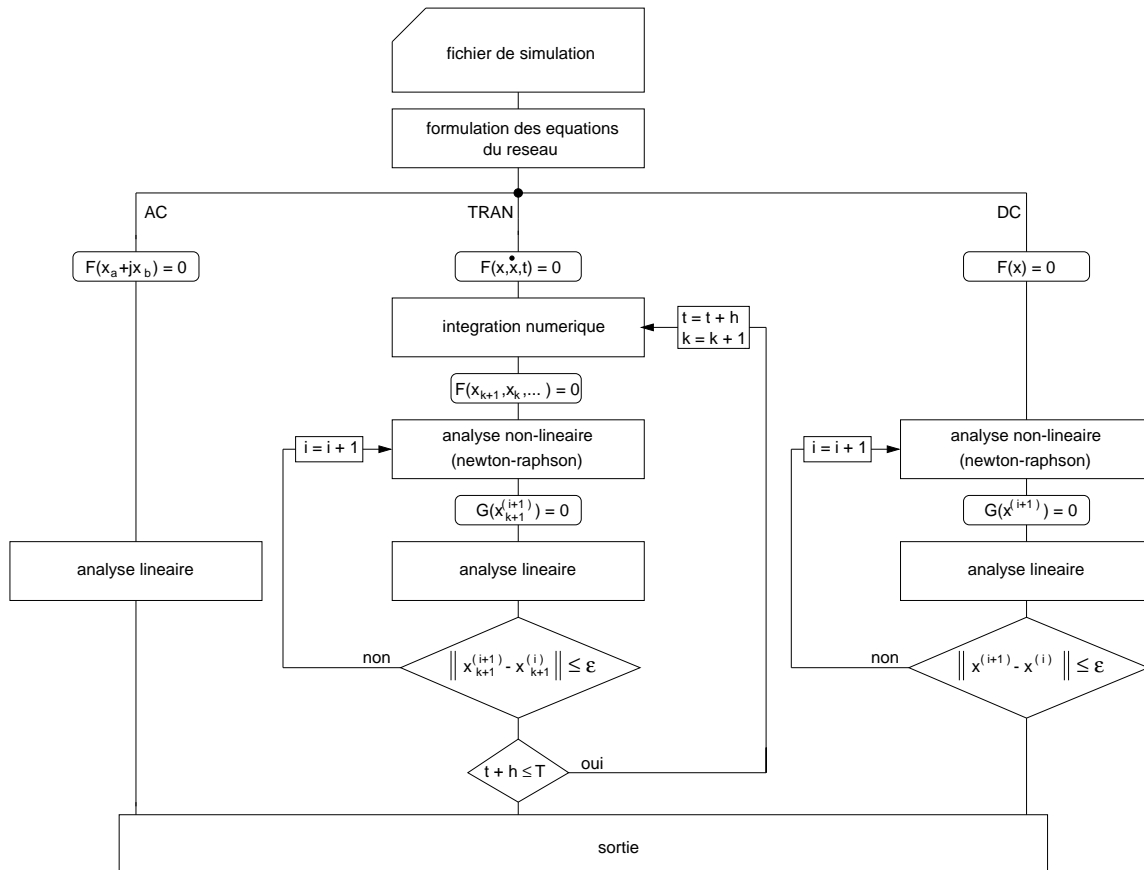


FIGURE 3 –

Représentation fonctionnelle d'un simulateur électrique standard

2.4.2 Erreur de méthode

L'erreur de méthode est liée au constat que le problème posé n'étant pas soluble analytiquement. Ainsi, on a recours à une méthode numérique présentant inévitablement, à chaque étape de calcul, une erreur d'arrondi par la taille limitée des nombres manipulés informatiquement. L'effet de ces erreurs d'arrondi sur le résultat final étant pratiquement proportionnel au nombre d'opérations élémentaires, il dépend de la méthode mise en œuvre. Ainsi pour comparer deux méthodes, on peut estimer le comptage du nombre d'opérations élémentaires effectuées selon la règle empirique : l'opération élémentaire étant l'addition (ou la soustraction), le temps nécessaire pour effectuer une multiplication est quatre fois plus important et le temps nécessaire pour effectuer une division est dix fois plus important.

2.4.3 Erreur de troncature

Les erreurs de troncature sont liées à la nature itérative de certaines méthodes numériques. Dans une méthode itérative, la solution y du problème à résoudre est la limite d'une suite y_k de solutions calculées approximativement, l'algorithme s'arrêtant après $N + 1$ itérations. On définit l'erreur de troncature locale (la LTE) comme l'erreur commise sur la solution calculée y_{k+1} , assumant que toutes les précédentes solutions $y_k, y_{k-1}, y_{k-2}, \dots$ sont exactes. La LTE est proportionnelle au pas de calcul. Quant à

l'erreur de troncature globale (la GTE) sur y_{N+1} elle dépend à la fois de la LTE sur y_{N+1} et des LTE précédentes. Une méthode numérique est dite stable si la contribution de la LTE sur la valeur y_k diminue lorsque N augmente, sinon elle est dite instable.

On peut noter qu'en règle générale, en diminuant le pas de calcul d'un algorithme on diminue l'erreur de troncature, mais en diminuant la vitesse de convergence, on risque de faire croître le nombre d'opérations élémentaires à effectuer, augmentant ainsi l'erreur de méthode.

2.5 Le conditionnement des méthodes

Il est rare qu'une méthode d'analyse numérique puisse être directement appliquée à la résolution d'un problème physique sans problème de précision, de dépassement des nombres machine ou de temps de calcul. La nature exponentielle des lois mathématiques modélisant certains dispositifs physiques, la très grande étendue possible des valeurs des différents composants (il n'est pas rare que dans un même circuit les résistances et les condensateurs soient chacun dans une gamme de valeur de 1 à 10^6), ..., font que l'électronique n'échappe pas à ce principe. Ainsi l'adoption des techniques de conditionnement des algorithmes, souvent heuristiques, est de règle en simulation électrique, et la fiabilité des simulateurs électriques standards actuels provient essentiellement de plus d'un quart de siècle de "réglages" des différents algorithmes.

2.6 Les limites d'utilisation d'un simulateur

De part les différentes erreurs de calculs associées aux différentes méthodes numériques et de part sa complexité, si un simulateur électrique est un outil de conception de circuits intégrés incontournable et puissant, c'est également un outil conditionnellement fiable, dont l'utilisation peut parfois s'avérer délicate et incertaine. L'utilisateur averti doit toujours pouvoir interpréter les résultats et si nécessaire affiner ou modifier les différents paramètres de réglage et de contrôle des algorithmes.

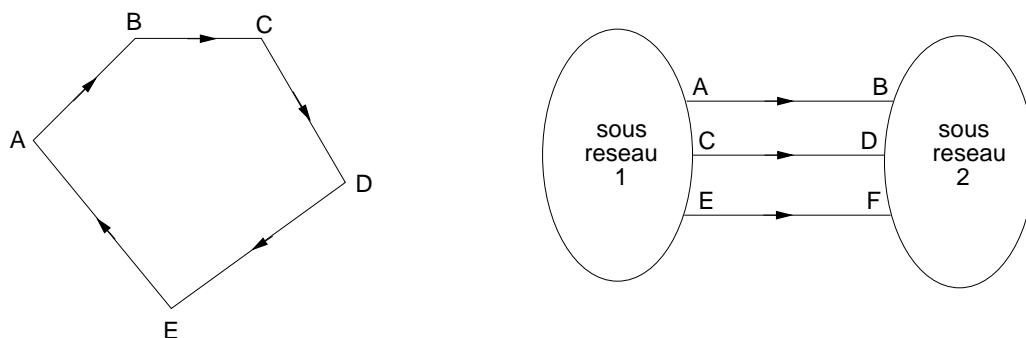
3 Méthodes de formulation des équations

Pour déterminer le système d'équations décrivant le circuit à analyser, ce dernier doit être préalablement représenté par un réseau de branches idéales interconnectées, chaque branche étant caractérisée par une relation mathématique appelée équation de constitution. D'autre part, pour pouvoir modéliser un circuit électrique, le système d'équations doit respecter deux contraintes : chaque équation de constitution et les lois de Kirchhoff en tension et en courant qui fixent les contraintes topologiques de connexion, doivent être respectées à tout instant, c'est à dire sur chaque pas de calcul.

3.1 Les lois de Kirchhoff

Les lois de Kirchhoff (*figure 4*) sont les deux axiomes fondamentaux de l'électricité. La loi de Kirchhoff en tension ou KVL, stipule qu'une boucle (maille) étant un ensemble de branches décrivant un parcours fermé, la somme des tensions instantanées des branches orientées dans le même sens que la boucle est nulle. La loi de Kirchhoff en courant ou KCL, stipule qu'une coupure étant l'ensemble des branches reliant deux parties disjointes d'un réseau, la somme des courants instantanés des branches constituant

une coupure et orientées dans le même sens c'est à dire du premier sous-réseau vers le second sous-réseau est nulle.



$$V_{AB}(t) + V_{BC}(t) + V_{CD}(t) + V_{DE}(t) + V_{EA}(t) = 0$$

$$I_{AB}(t) + I_{CD}(t) + I_{EF}(t) = 0$$

FIGURE 4 –

Les lois de Kirchhoff en tension et en courant

3.2 L'équation de constitution généralisée

Une branche d'un réseau électrique (*figure 5*) est entièrement caractérisée par un courant I_k et une tension U_k , telle que pour une branche du réseau définie en tension, l'équation de constitution peut se formuler mathématiquement par l'expression

$$U_k = v_k + Z_k(I_k - j_k)$$

dans laquelle v_k et j_k sont des sources de tension et de courant indépendantes ou commandées, alors que pour une branche définie en courant l'équation de constitution est fournie par l'expression duale

$$I_k = j_k + Y_k(U_k - v_k).$$

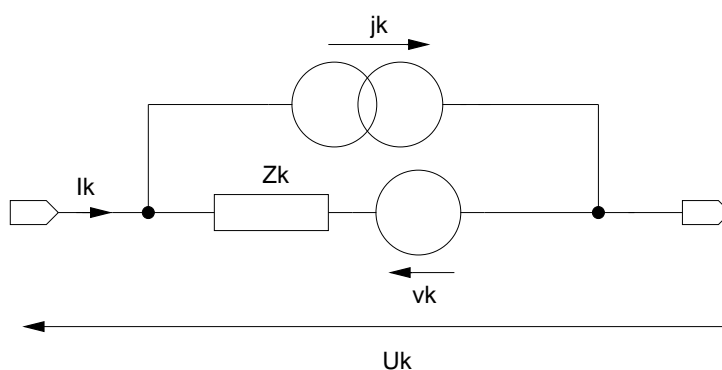


FIGURE 5 –

L'équation de constitution généralisée

3.3 L'analyse nodale

3.3.1 Les différentes méthodes de formulation des équations

Il existe de nombreuses méthodes de formulation permettant une mise en équation automatique, c'est à dire sans possibilité d'oubli et de double emploi [?],[?],[?],[?]. Toutes ces méthodes peuvent être regroupées en deux familles distinctes [?]. La première famille cherche une mise en équation en imposant la mise forme des équations en fonction de critères théoriques : d'après les lois de Kirchhoff, un réseau électrique composé de B branches et $N + 1$ nœuds, possède B équations de branche, N équations de nœud sur les courants et ainsi $B - N$ équations sur les tensions. Le nombre d'équations à résoudre est donc égal à N ou $B - N$ selon la méthode utilisée. Si généralement, le nombre d'équations est faible, le défaut majeur de ces méthodes est qu'elles conduisent à mettre les équations de constitution sous la forme d'une matrice impédance ou admittance qui peut ne pas exister (figure 6 et table 1).

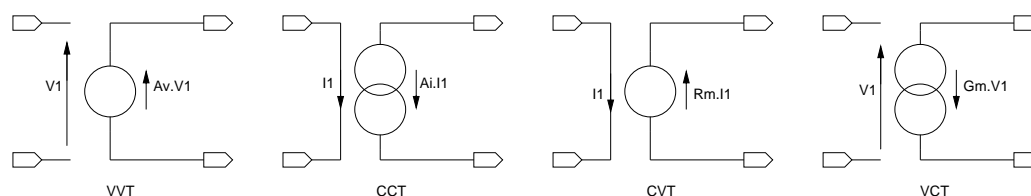


FIGURE 6 –

Les quatre transducteurs basiques

type	$[Z] = \begin{pmatrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{pmatrix}$	$[Y] = \begin{pmatrix} Y_{11} & Y_{12} \\ Y_{21} & Y_{22} \end{pmatrix}$	$[T] = \begin{pmatrix} A & B \\ C & D \end{pmatrix}$
VVT	?	?	$\begin{pmatrix} 1/Av & 0 \\ 0 & 0 \end{pmatrix}$
CCT	?	?	$\begin{pmatrix} 0 & 0 \\ 0 & 1/Ai \end{pmatrix}$
VCT	?	$\begin{pmatrix} 0 & 0 \\ Gm & 0 \end{pmatrix}$	$\begin{pmatrix} 0 & 1/Gm \\ 0 & 0 \end{pmatrix}$
CVT	$\begin{pmatrix} 0 & 0 \\ Rm & 0 \end{pmatrix}$?	$\begin{pmatrix} 0 & 0 \\ 1/Rm & 0 \end{pmatrix}$

TABLE 1 –

Coefficients des matrices impédance, admittance et transfert des quatre transducteurs basiques

La seconde famille, se fonde sur des critères purement numériques, les équations du circuit sont formulées en termes de tensions de nœud, de tensions de branche, de courants de branche, de charges capacitatives et de flux inductifs, pour former un tableau (ou une liste), et on "laisse" au programme d'analyse numérique le soin de résoudre au mieux le système. On échappe ainsi aux problèmes des éventuelles inexistentances des matrices impédance ou admittance, mais on génère des matrices de taille très importante mais

heureusement très creuse (beaucoup de termes nuls) imposant l'utilisation de techniques de calcul spécifiques et un conditionnement préalable du système d'équations, lourd à mettre en œuvre [?].

Bien qu'appartenant à la première famille des méthodes de formulation des équations, l'analyse nodale est très largement utilisée dans les simulateurs modernes, du fait de la simplicité de programmation. En outre, le problème d'inexistence éventuelle de matrices admittance et l'impossibilité de gérer les sources de tension idéales ont été levés par l'analyse nodale modifiée.

3.3.2 La matrice admittance indéfinie

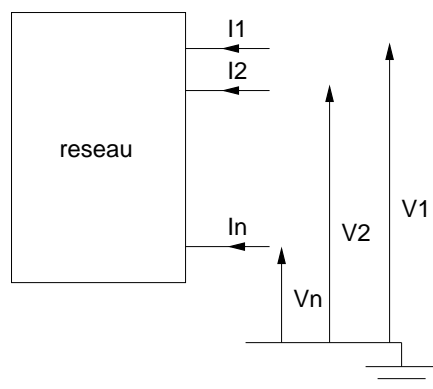


FIGURE 7 –

Réseau flottant à entrée multiples

Si nous considérons le circuit électrique de la *figure 7*, par construction, il est tel que

1. $\sum_{k=1}^n I_k = 0$ de part la loi de Kirchhoff en courant
2. Chaque courant I_k dépend de la différence de potentiel entre le K^{ieme} terminaison et le nœud de référence.

Ainsi, il est caractérisé par la représentation matricielle

$$[I] = [Y][V]$$

avec

$$[Y] = \begin{pmatrix} y_{11} & y_{12} & \dots & y_{1n} \\ y_{21} & y_{22} & \dots & y_{2n} \\ \vdots & \vdots & & \vdots \\ y_{n1} & y_{n2} & \dots & y_{nn} \end{pmatrix}$$

La matrice $[Y]$ est appelée matrice admittance indéfinie par le fait qu'aucun accès du réseau considéré n'a été pris comme référence. Cette matrice peut systématiquement être écrite par inspection du circuit avec :

pour les éléments diagonaux :

Y_{11} est la somme des admittances connectées au nœud 1

Y_{22} est la somme des admittances connectées au nœud 2

\vdots
 Y_{nn} est la somme des admittances connectées au nœud n,

pour les éléments non diagonaux tous affectés du signe moins :

Y_{12} est la somme des admittances connectées entre le nœud 1 et le nœud 2

Y_{13} est la somme des admittances connectées entre le nœud 1 et le nœud 3

\dots

Y_{1n} est la somme des admittances connectées entre le nœud 1 et le nœud n

\vdots

Y_{21} est la somme des admittances connectées entre le nœud 2 et le nœud 1

Y_{23} est la somme des admittances connectées entre le nœud 2 et le nœud 3

\dots

Y_{2n} est la somme des admittances connectées entre le nœud 2 et le nœud n

\vdots

Ainsi, le circuit de la *figure 8* est caractérisé par la matrice admittance indéfinie

$$[Y] = \begin{pmatrix} Y_1 + Y_4 + Y_6 & -Y_6 & -Y_4 & -Y_1 \\ -Y_6 & Y_3 + Y_5 + Y_6 & -Y_5 & -Y_3 \\ -Y_4 & -Y_5 & Y_2 + Y_4 + Y_5 & -Y_2 \\ -Y_1 & -Y_3 & -Y_2 & Y_1 + Y_2 + Y_3 \end{pmatrix}.$$

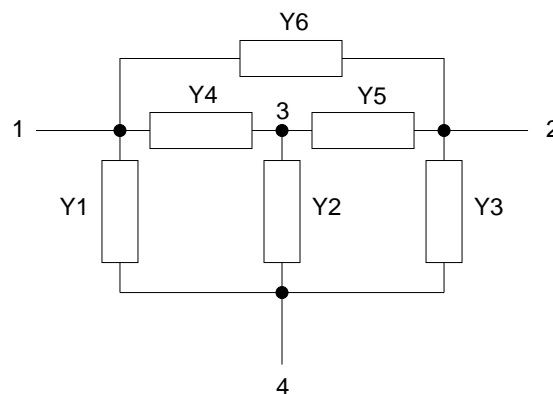


FIGURE 8 –

Un exemple d'analyse nodale par la matrice admittance indéfinie

De part sa nature flottante, la matrice admittance indéfinie est un outil très utile de la théorie des réseaux [?],[?], elle est facilement manipulable et elle possède de nombreuses propriétés analytiques et topologiques :

● Quelques propriétés analytiques de la matrice admittance indéfinie :

1. la somme des éléments de chaque colonne est nulle
2. la somme des éléments de chaque ligne est nulle
3. le déterminant de la matrice est nul
4. tous les premiers cofacteurs sont égaux.

Les deux premières propriétés font que la matrice admittance indéfinie est une matrice dite à sommes nulles, ce qui entraîne les deux autres propriétés.

● Quelques propriétés topologiques de la matrice admittance indéfinie :

1. connexion parallèle :

la matrice admittance indéfinie de plusieurs multipôles en parallèle est simplement obtenue en additionnant les éléments correspondants des matrices individuelles, toutes les matrices doivent être du même ordre, et on notera que l'ordre d'une matrice peut être incrémenté en ajoutant une ligne et une colonne de zéros

2. transformation en matrice admittance définie :

si un accès quelconque est référencé, la matrice indéfinie est transformée en matrice définie (usuellement simplement matrice admittance) en supprimant la ligne et la colonne du nœud référencé

3. transformation en matrice admittance indéfinie :

inversement, partant d'une matrice admittance définie et d'un accès référencé, la matrice peut être rendue flottante en ajoutant une ligne et une colonne de telle sorte que la somme des éléments des lignes et des colonnes soient nuls.

3.3.3 La matrice admittance définie

Bien entendu, la matrice admittance définie peut également être synthétisée systématiquement par inspection, ainsi le circuit de la *figure 9* peut directement être mis en équation pour une analyse en régime sinusoïdal par

$$\begin{pmatrix} I_1 \\ 0 \\ V_7 \cdot G_6 \\ 0 \end{pmatrix} = [Y] \begin{pmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{pmatrix}$$

et avec

$$[Y] = \begin{pmatrix} G_1 + \frac{1}{jL_2\omega} & -G_1 - \frac{1}{jL_2\omega} & 0 & 0 \\ -G_1 - \frac{1}{jL_2\omega} & G_1 + G_3 + \frac{1}{jL_2\omega} + \frac{1}{jL_4\omega} & -\frac{1}{jL_4\omega} & 0 \\ 0 & -\frac{1}{jL_4\omega} & G_6 + jC_5\omega + \frac{1}{jL_4\omega} & -jC_5\omega \\ 0 & 0 & -jC_5\omega & G_7 + jC_5\omega \end{pmatrix}.$$

L'examen de la matrice admittance appelle quelques observations :

1. le circuit étant purement passif la matrice est symétrique
2. la source de courant idéal est considérée directement
3. la source de tension avec sa résistance série a été transformée en source de courant par transformation Thévenin-Norton (*figure 10*), il faut noter qu'une source de tension idéale (élément purement mathématique sans réalité physique) aurait nécessité l'adjonction d'une résistance série arbitraire mais très petite devant les autres résistances du réseau.
4. plus le réseau est topologiquement transversal, plus la matrice est creuse, ceci est d'ailleurs une caractéristique essentielle des circuits électroniques comportant des éléments actifs, puisque un terme non diagonal traduit un couplage entre deux nœuds et qu'un tel couplage est d'autant plus rare que les nœuds sont éloignés pour des considérations évidente de contrôle de la stabilité.

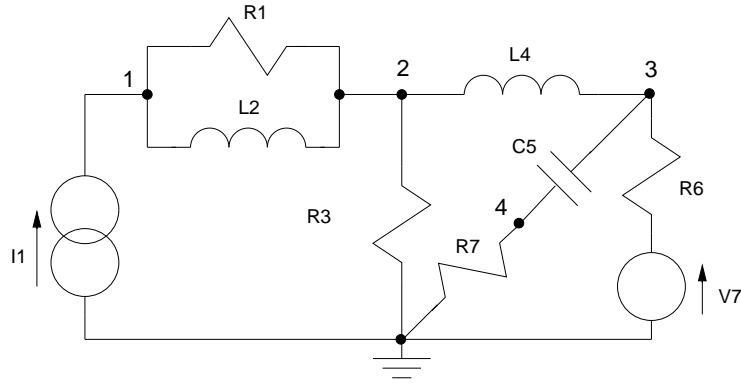


FIGURE 9 -

Un exemple d'analyse nodale par la matrice admittance définie

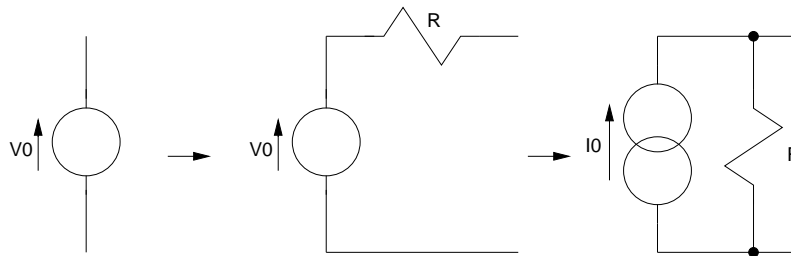


FIGURE 10 -

Transformation Thévenin-Norton d'une source de tension idéale : $I_0 = V_0/R$

3.3.4 Le traitement des sources contrôlées

L'inexistence de matrices admittances pour les transducteurs tension-tension (VVT), courant-tension (CVT) et courant-courant (CCT) peut être levée par l'ajout de résistances, afin de rendre ces transducteurs non idéaux mais traitables par l'analyse nodale. Ainsi, la source de tension contrôlée en courant (CVT) non-idéale de la *figure 11* admet la matrice admittance définie

$$[Y_{CVT}] = \begin{pmatrix} G_e & 0 \\ -R_m G_e G_s & G_s \end{pmatrix}.$$

Eventuellement, au prix d'une complexité encore plus grande en terme de nombre de nœuds, un raffinement supplémentaire peut être obtenu en considérant le circuit "corrigé" de la *figure 11*.

3.3.5 Un exemple de "manipulation" matricielle

Considérons le circuit actif de la *figure 12*, constitué d'un réseau d'éléments passifs et d'un amplificateur opérationnel modélisé par le schéma de la *figure 13*.

La matrice admittance indéfinie du réseau passif peut être écrit par inspection avec

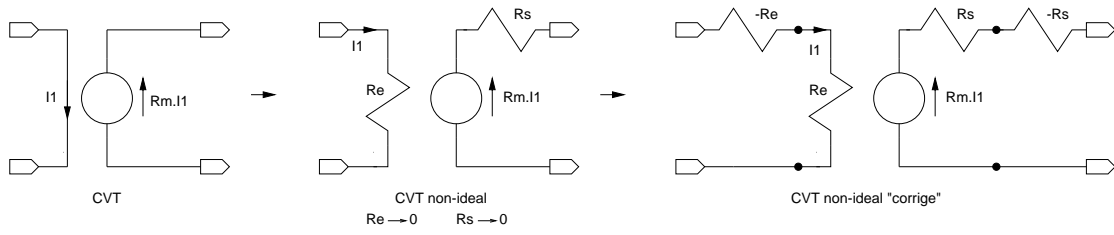


FIGURE 11 -

Traitements d'une source de tension contrôlée en courant

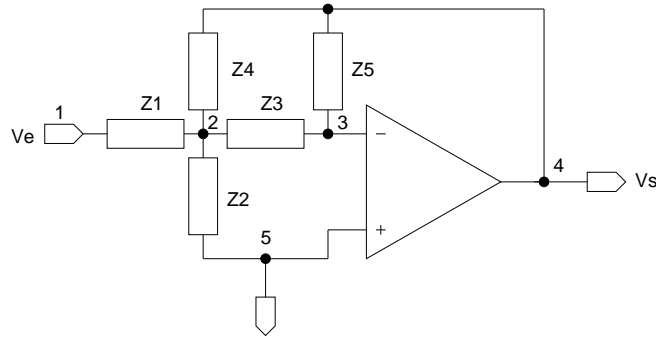


FIGURE 12 -

Un circuit actif

$$[Y_a] = \begin{pmatrix} Y_1 & -Y_1 & 0 & 0 & 0 \\ -Y_1 & Y_1 + Y_2 + Y_3 + Y_4 & -Y_3 & -Y_4 & -Y_2 \\ 0 & -Y_3 & Y_3 + Y_5 & -Y_5 & 0 \\ 0 & -Y_4 & -Y_5 & Y_4 + Y_5 & 0 \\ 0 & -Y_2 & 0 & 0 & Y_2 \end{pmatrix},$$

de même la matrice admittance définie de l'amplificateur opérationnel s'écrit

$$[Y_{aop}] = \begin{pmatrix} Ge & -Ge & 0 \\ -Ge & Ge & 0 \\ A_0Gs & -A_0Gs & Gs \end{pmatrix}.$$

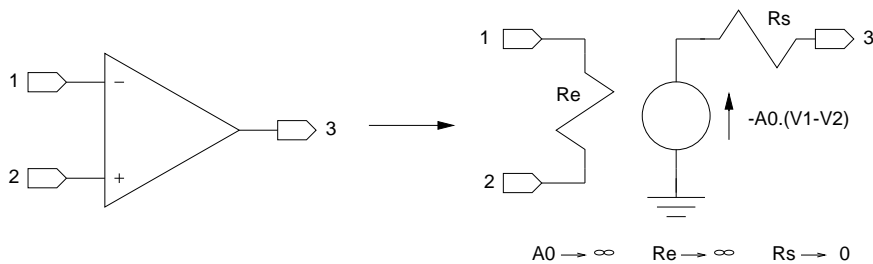


FIGURE 13 -

Modélisation de l'AOP

Pour pouvoir être mis en parallèle avec le circuit passif, on doit dans un premier

temps transformer la matrice définie de l'AOP à trois nœuds en matrice définie à deux nœuds en supprimant la deuxième ligne et la deuxième colonne, puis dans un deuxième temps transformer cette matrice définie en une matrice indéfinie et enfin ajouter deux accès supplémentaires. On arrive ainsi à la matrice admittance indéfinie

$$[Y_b] = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & Ge & 0 & -Ge \\ 0 & 0 & A_0Gs & Gs & -(A_0 + 1)Gs \\ 0 & 0 & -Ge - A_0Gs & -Gs & Ge + Gs + A_0Gs \end{pmatrix},$$

et à la matrice admittance indéfinie du circuit actif complet

$$[Y] = \begin{pmatrix} Y_1 & -Y_1 & 0 & 0 & 0 \\ -Y_1 & Y_1 + Y_2 + Y_3 + Y_4 & -Y_3 & -Y_4 & -Y_2 \\ 0 & -Y_3 & Y_3 + Y_5 + Ge & -Y_5 & -Ge \\ 0 & -Y_4 & -Y_5 + A_0Gs & Y_4 + Y_5 + Gs & -(A_0 + 1)Gs \\ 0 & -Y_2 & -Ge - A_0Gs & -Gs & Y_2 + Ge + Gs + A_0Gs \end{pmatrix}.$$

3.4 L'analyse nodale modifiée (la MNA)

L'analyse nodale modifiée [?], permet de s'affranchir directement du problème de l'impossibilité de gérer les sources de tension idéales et de l'inexistence éventuelle de matrices admittances. Pour ce faire, la matrice MNA est exprimée sous la forme

$$\begin{pmatrix} [Y_R] & [B] \\ [C] & [D] \end{pmatrix} \begin{pmatrix} \vec{V} \\ \vec{J} \end{pmatrix} = \begin{pmatrix} \vec{I} \\ \vec{F} \end{pmatrix}.$$

La construction du système d'équations est un peu compliqué [?]. La matrice $[Y_R]$ est la matrice nodale réduite excluant les contributions dues aux sources de tension et aux éléments contrôlés en courant. Le vecteur \vec{V} est le vecteur tension de nœud, \vec{J} est le vecteur courant des branches définies en tension, \vec{F} est le vecteur tension des branches définies en tension, la matrice $[Y_R]$ et le vecteur \vec{I} sont construits comme pour l'analyse nodale. Si il n'y a pas de branche définie en courant avec des courants commandés, la matrice $[B]$ est construite avec des zéros et des uns tels que

$$\begin{aligned} b_{ij} &= 0 \text{ si le nœud } i \text{ n'est pas connecté à la branche } j \\ b_{ij} &= +1 \text{ si le nœud } i \text{ est le nœud positif de la la branche } j \\ b_{ij} &= -1 \text{ si le nœud } i \text{ est le nœud négatif de la la branche } j \end{aligned}$$

Dans le cas contraire, la matrice $[B]$ devra contenir les coefficients de couplage fixant les différentes relations. Quant aux matrices $[C]$ et $[D]$ et au vecteur \vec{F} ils contiennent les relations de branches des p branches définies en tension. Ainsi, l'analyse MNA du circuit de la *figure 14* conduit au système

$$\left(\begin{pmatrix} G_2 & -G_2 & 0 \\ -G_2 & G_2 + G_3 + j(C_4 + C_5)\omega & -jC_5\omega \\ 0 & g_6 - jC_5\omega & G_7 + jC_5\omega \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \right) \begin{pmatrix} V_1 \\ V_2 \\ V_3 \\ I_1 \end{pmatrix} =$$

$$\left(\begin{array}{c} \left(\begin{array}{c} 0 \\ 0 \\ 0 \end{array} \right) \\ E \end{array} \right)$$

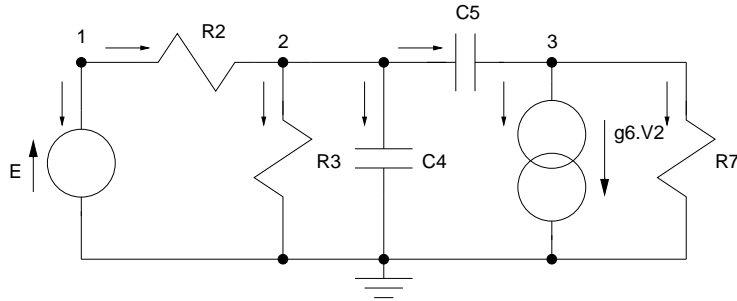


FIGURE 14 –

Un exemple de circuit pour l'analyse nodale modifiée [?]

4 Méthodes de résolution d'un système d'équations linéaires

Tous les types d'analyse requièrent la résolution de l'équation matricielle

$$[A] \vec{X} = \vec{B}.$$

La résolution numérique peut être effectuée itérativement, dans ce cas elle peut typiquement être vue comme un problème de minimisation, c'est à dire de détermination du vecteur \vec{X} tel que le scalaire

$$r = ([A]\vec{X} - \vec{B})^T([A]\vec{X} - \vec{B})$$

soit minimum. Si dans quelque cas cette technique peu donner des résultats satisfaisants, dans beaucoup d'autres les différents algorithmes utilisables (Gauss-Seidel, Fletcher-Powell,..) peuvent présenter des problèmes de convergence. Ainsi, les techniques de résolution utilisées en simulation électrique de deuxième génération sont des méthodes de résolutions directes.

Une solution directe de l'équation matricielle peut être obtenue par inversion de la matrice $[A]$ avec

$$\vec{X} = [A]^{-1}\vec{B} \text{ et } [A]^{-1} = \frac{adj[A]}{det[A]}.$$

Pour ce faire, on doit effectuer de l'ordre de $(n + 1)!$ additions, multipliées par $(n + 2)!$ multiplications, multipliées par n divisions, correspondant à un nombre d'opérations élémentaires de $2 \cdot 10^9$ pour une matrice d'ordre 10. La quantité d'opérations à effectuer pouvant devenir astronomique lorsque le nombre d'équations approche la centaine, cette méthode n'est jamais utilisée en pratique. Les deux procédures de calcul principalement utilisées en simulation électrique sont des procédures dites d'élimination [?]. Dans le

principe, on détermine une matrice $[M]$ inversible telle que la matrice $[M][A]$ soit triangulaire supérieure (ou inférieure) et on résoud le système linéaire

$$[M][A]\vec{X} = [M]\vec{B}$$

par l'algorithme dit de la remontée.

4.1 L'algorithme de la remontée

Si une matrice $[A]$ est triangulaire supérieure (ou inférieure) la solution du système $[A]\vec{X} = \vec{B}$ est directe. En effet, ayant

$$\begin{array}{ccccccc} a_{11}x_1 + & \dots & + a_{1,n-1}x_{n-1} & + a_{1,n}x_n & = & b_1 \\ & \ddots & & \vdots & & \vdots \\ & & \ddots & \vdots & & \vdots \\ & & & a_{n-1,n-1}x_{n-1} & + a_{n-1,n}x_n & = & b_{n-1} \\ & & & & a_{nn}x_n & = & b_n \end{array}$$

par "remontée" de la valeur x_n la solution est directe puisque

$$\begin{array}{l} x_n = b_n/a_{nn} \\ x_{n-1} = (b_{n-1} - a_{n-1,n}x_n)/a_{n-1,n-1} \\ \vdots \\ x_1 = (b_1 - a_{12}x_2 - \dots - a_{1,n-1}x_{n-1} - a_{1,n}x_n)/a_{11} \end{array}$$

L'algorithme de la remontée nécessite $\frac{n(n-1)}{2}$ additions, $\frac{n(n-1)}{2}$ multiplications et n divisions.

4.2 L'algorithme d'élimination de Gauss

Soit le système de n équations linéaires à n inconnues

$$\begin{array}{ccccccc} a_{11}x_1 + a_{12}x_2 + \dots + a_{1,n}x_n & = & b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2,n}x_n & = & b_2 \\ & & \vdots & \vdots & \vdots \\ a_{n,1}x_1 + a_{n,2}x_2 + \dots + a_{n,n}x_n & = & b_n \end{array}$$

Avec a_{11} différent de 0 (terme de pivot), éliminons x_1 à partir de la seconde équation en ajoutant à celle-ci la première multipliée par $-a_{21}/a_{11}$, à la troisième la première multipliée par $-a_{31}/a_{11}$, ..., etc. On arrive ainsi au nouveau système

$$\begin{aligned}
a_{11}x_1 + a_{12}x_2 + \dots + a_{1,n}x_n &= b_1 \\
a_{22}^{(1)}x_2 + \dots + a_{2,n}^{(1)}x_n &= b_2^{(1)} \\
&\vdots \\
a_{n,2}^{(1)}x_2 + \dots + a_{n,n}^{(1)}x_n &= b_n^{(1)}
\end{aligned}$$

avec $a_{ij}^{(1)} = a_{ij} - \frac{a_{i1}}{a_{11}}a_{1j}$, $b_i^{(1)} = b_i - \frac{a_{i1}}{a_{11}}b_1$ pour $i, j = 2, \dots, n$.

Supposons que le pivot $a_{22}^{(1)}$ soit différent de zéro, et éliminons x_2 des $n - 2$ dernières équations, nous arrivons au système

$$\begin{aligned}
a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \dots + a_{1,n}x_n &= b_1 \\
a_{22}^{(1)}x_2 + a_{23}^{(1)}x_3 + \dots + a_{2,n}^{(1)}x_n &= b_2^{(1)} \\
a_{33}^{(2)}x_3 + \dots + a_{3,n}^{(1)}x_n &= b_3^{(2)} \\
&\vdots \\
a_{n,3}^{(2)}x_3 + \dots + a_{n,n}^{(2)}x_n &= b_n^{(2)}
\end{aligned}$$

avec $a_{ij}^{(2)} = a_{ij}^{(1)} - \frac{a_{i2}^{(1)}}{a_{22}^{(1)}}a_{2j}^{(1)}$, $b_i^{(2)} = b_i^{(1)} - \frac{a_{i2}^{(1)}}{a_{22}^{(1)}}b_2^{(1)}$ pour $i, j = 3, \dots, n$.

En poursuivant la procédure d'élimination, on passe en $n - 1$ étapes au système triangulaire supérieur

$$\begin{aligned}
a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \dots + a_{1,n}x_n &= b_1 \\
a_{22}^{(1)}x_2 + a_{23}^{(1)}x_3 + \dots + a_{2,n}^{(1)}x_n &= b_2^{(1)} \\
a_{33}^{(2)}x_3 + \dots + a_{3,n}^{(1)}x_n &= b_3^{(2)} \\
&\vdots \\
a_{n,n}^{(n-1)}x_n &= b_n^{(n-1)}
\end{aligned}$$

L'algorithme de Gauss est facilement programmable et nécessite $\frac{n^3-n}{3}$ additions, $\frac{n^3-n}{3}$ multiplications et $\frac{n^2-n}{2}$ divisions. Ainsi compte tenu de la remontée, l'inversion d'une matrice de $n = 10$ se fait en 2200 opérations élémentaires (à comparer au $2 \cdot 10^9$ de la première méthode). La méthode d'élimination de Gauss est une méthode très classique de résolution d'un système d'équations linéaires.

4.3 La décomposition LU

Cette méthode de résolution est directement inspirée de la méthode de Gauss. Dans ce cas la matrice $[A]$ est décomposée en une matrice triangulaire inférieure L (**L**ower) et une matrice triangulaire supérieure U (**U**pper) telles que le système à résoudre soit de la forme

$$[A] \vec{X} = [L][U]\vec{X} = \vec{B}.$$

Ensuite, l'algorithme de la remontée nous fournit

$$\vec{Y} = [U]\vec{X} = [L]^{-1}\vec{B},$$

puis l'algorithme de la redescente (algorithme de la remontée inversé) nous permet d'obtenir \vec{X} avec

$$\vec{X} = [U]^{-1}\vec{Y} = [U]^{-1}[L]^{-1}\vec{B}.$$

Il y a différentes méthodes pour obtenir les matrices $[L]$ et $[U]$ [?], et termes de nombre d'opérations élémentaires la décomposition LU est équivalente à la méthode de Gauss. Son intérêt majeur est que dans le cas de plusieurs vecteurs d'excitation, elle offre la possibilité de minimiser le nombres d'opérations à effectuer [?] [?].

4.4 Le conditionnement des méthodes

Un des principaux problème posé par l'algorithme d'élimination, c'est qu'il a besoin d'effectuer une division par le pivot. Ce pivot devant être non nul et autant que faire ce peut, pas trop petit, une phase préalable de permutation de lignes et de colonnes est nécessaire pour optimiser numériquement la procédure. D'autre part, lors de la décomposition des matrices, certains termes nuls de la matrice initiale peuvent devenir non nuls et la matrice se remplit. Différents techniques de conditionnement jouant sur le choix du pivot par une réorganisation des lignes et des colonnes doivent être utilisées pour conserver le caractère creux de la matrice initiale [?]. Enfin, une des spécificité de l'analyse nodale est de générer des matrices pratiquement symétriques, et l'effort en termes de nombre d'opérations pour symétriser la matrice est souvent faible pour un gain global important. Ce faisant, le conditionnement des méthodes de formulation des équations est du point de vue informatique une procédure qui peut s'avérée assez coûteuse en terme de temps de calcul et fortement dépendante du type de procédure utilisé.

5 Méthodes numériques d'intégration

Intégrer numériquement une équation différentielle consiste à partir de la valeur initiale x_0 et calculer successivement les valeurs $x(t_1), x(t_2), \dots, x(t_k), \dots$, jusqu'à la valeur finale $x(t_N)$ correspondant à la solution recherchée. En simulation électrique, l'intégration numérique transforme un système d'équations différentielles en un système d'équations algébriques, en d'autres termes il transforme un réseau dynamique éventuellement non-linéaire en un réseau "résistif" éventuellement non-linéaire.

5.1 Les méthodes d'intégration d'Euler explicite et implicite

Considérons le circuit RC de la *figure 15* et écrivons les lois de Kirchhoff correspondantes, la variable $x(t)$ étant la tension aux bornes de la résistance et du condensateur, on obtient

$$\frac{1}{R}x(t) + C\frac{dx(t)}{dt} = 0$$

soit

$$\frac{dx(t)}{dt} = -\frac{1}{RC}x(t)$$

ou

$$\dot{x}(t) = -\lambda x(t).$$

La constante x_0 étant la tension initiale, le circuit RC a pour solution analytique

$$x = x_0 e^{-\lambda t}.$$

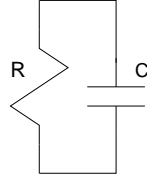


FIGURE 15 –

Circuit RC du premier ordre

Si on veut résoudre numériquement le circuit, au voisinage du point t_{k+1} , $h = t_{k+1} - t_k$ étant le pas de calcul choisi arbitrairement identique pour tous les instants t_k , et en notant par convenance $x(t_{k+i}) = x_{k+i}$ on peut effectuer le développement en série de Taylor de $x(t)$

$$x_{k+1} = x_k + h \dot{x}_k + \frac{h^2}{2} \frac{d^2 x}{dt^2}(\zeta)$$

et tronquer la série (le terme noté en dérivé correspond à l'erreur de troncature locale) pour obtenir la formule d'intégration d'Euler explicite ou FEDI (**F**orward **E**uler **D**iscrete **I**ntegrator)

$$x_{k+1} = x_k + h \dot{x}_k.$$

Ainsi, partant d'une représentation des équations du circuit sous forme normale

$$\dot{x}(t) = f(x, t),$$

par discrétisation à l'instant t_k

$$\dot{x}_k = f(x_k, t_k),$$

on obtient directement par extrapolation la valeur intermédiaire

$$x_{k+1} = x_k (1 - h\lambda)$$

soit après $N + 1$ pas de calcul la solution numérique finale

$$x_N = x_0 (1 - h\lambda)^N.$$

On peut également développer la dérivée en série de Taylor

$$\dot{x}_{k+1} = \dot{x}_k + h \frac{d^2 x}{dt^2}(\zeta).$$

pour obtenir

$$x_{k+1} = x_k + h \dot{x}_{k+1} - \frac{h^2}{2} \frac{d^2 x}{dt^2}(\zeta),$$

ce qui donne après troncature, la formule d'intégration d'Euler implicite ou BEDI (**B**ackward **E**uler **D**iscrete **I**ntegrator)

$$x_{k+1} = x_k + h \dot{x}_{k+1}.$$

et

$$x_{k+1} = \frac{x_k}{1+h\lambda}$$

soit après $N + 1$ pas de calcul

$$x_N = x_0 \left(\frac{1}{1+h\lambda} \right)^N.$$

Si en terme de complexité de calcul la méthode d'intégration explicite est plus simple, pour les valeurs de $\lambda > 0$ conduisant à une solution analytiquement stable c'est à dire telles que $x(t) \rightarrow 0$ lorsque $t \rightarrow \infty$:

1. la méthode d'intégration explicite n'est stable et conduit à la solution recherchée qu'avec $0 \leq h \leq 1/\lambda$
2. la méthode d'intégration implicite est toujours stable et conduit à la solution recherchée quelque soit la valeur de λ .

Ceci est une caractéristique des méthodes numériques d'intégration. En règle générale, les méthodes implicites son inconditionnellement stables, alors que les les méthodes explicites ne le sont que conditionnellement, obligeant de choisir un pas d'intégration h tel $h < 1/\lambda_{max}$, $1/\lambda_{max}$ étant la plus petite constante de temps du circuit à analyser. Or, les circuits électroniques analogiques présentent très souvent des pôles très hautes fréquences (typiquement pôles parasites associés à de faibles valeurs de condensateurs parasites) qui sont connus pour intervenir de manière négligeable sur les différentes réponses parce que pratiquement ils ne seront pas excités ou que les transitoires correspondant sont très brefs. Sachant, que le temps d'observation de la réponse temporelle est généralement de l'ordre de $T = 1/\lambda_{min}$, les temps de calcul proportionnels à $\lambda_{max}/\lambda_{min}$ peuvent facilement devenir prohibitifs. Ainsi, pour la simulation électrique des circuits analogiques, les méthodes numériques d'intégration sont généralement des méthodes implicites. Le problème est différent pour les circuits numériques. Toutes les constantes de temps associées étant du même ordre de grandeur et les équations de réseau étant données sous forme normale, une méthode d'intégration explicite peut être efficacement utilisée. On peut noter qu'un circuit avec des constantes de temps inférieures au pas d'intégration est un circuit dit "relâché".

5.2 La méthode d'intégration trapézoïdale

En considérant le développement en série de Taylor de x

$$x_{k+1} = x_k + h \dot{x}_k + \frac{h^2}{2} \ddot{x}_k + \frac{h^3}{6} \frac{d^3x}{dt^3}(\zeta)$$

et avec

$$\dot{x}_{k+1} = \dot{x}_k + h \ddot{x}_k + \frac{h^2}{2} \frac{d^3x}{dt^3}(\zeta)$$

on détermine à l'erreur de troncature locale près, la formule de la méthode d'intégration trapézoïdale

$$x_{k+1} = x_k + \frac{h}{2}(\dot{x}_{k+1} + \dot{x}_k).$$

Ainsi, pour le circuit RC test de la *figure 15*, on obtient

$$x_{k+1} = x_k \frac{1 - h \lambda/2}{1 + h \lambda/2}$$

et

$$x_N = x_0 \left(\frac{1 - h \lambda/2}{1 + h \lambda/2} \right)^N.$$

Sous les mêmes conditions que la méthode d'Euler implicite, l'intégration numérique trapézoïdale est inconditionnellement stable. Une particularité notable de la méthode est que pour des valeurs de $h\lambda > 2$, x_k est positif pour les valeurs paires de k et négatif pour les valeurs impaires ainsi, pour contenir la propension de la méthode à osciller lorsque x_k approche 0, on devra veiller à maintenir l'erreur de troncature locale à un niveau cohérent. On notera que beaucoup de simulateurs électriques impose par défaut l'intégration trapézoïdale.

5.3 Les méthodes d'intégration de Gear

Contrairement aux méthodes d'intégration précédentes, la valeur de x_{k+1} est déterminée en fonction de \dot{x}_{k+1} , x_k et des valeurs précédentes x_{k-1} , x_{k-2} , ... Au premier ordre la méthode de Gear est identique à la méthode d'Euler, et à l'ordre 2, Gear2 s'écrit

$$x_{k+1} = \frac{4}{3}x_k - \frac{1}{3}x_{k-1} + \frac{2h}{3} \dot{x}_{k+1} - \frac{2h^3}{9} \frac{d^3x}{dt^3}(\zeta),$$

soit à l'erreur de troncature près

$$x_{k+1} = \frac{4}{3}x_k - \frac{1}{3}x_{k-1} + \frac{2h}{3} \dot{x}_{k+1}.$$

Pour les ordres inférieurs à deux, la méthode est inconditionnellement stable sinon elle est conditionnellement stable. Les méthodes de Gear sont des méthodes multipoints qui nécessitent la mémorisation des points de calcul précédents. D'autre part, elle doivent être déclenchées dans le sens qu'elle ne peuvent pas être calculées pour le premier point. Pour ce faire la technique généralement adoptée est de prendre la méthode d'Euler (Gear1) pour le premier point, Gear2 pour le deuxième point (, ... éventuellement) [?].

5.4 La traduction des méthodes d'intégration en termes de pôles

L'utilisation d'une méthode numérique d'intégration induit inévitablement un artefact de simulation dépendant des caractéristiques électriques du circuit simulé. Tous ces artefacts sont visibles dans le plan des fréquences par le fait qu'une méthode d'intégration effectue une transformation du plan de la variable complexe $P = \sigma + j\omega$ vers le plan de la variable $Z = \exp(ph)$. En effet, pour intégrer numériquement, il faut discrétiser donc échantillonner à intervalle régulier h une fonction temps continu $x(t)$ donnée. Cette fonction échantillonnée c'est à dire multipliée par un peigne de Dirac s'écrivant

$$x^*(t) = x(t) \sum_{-\infty}^{+\infty} \delta(t - nh) = \sum_{-\infty}^{+\infty} x(nh) \delta(t - nh),$$

par définition, on obtient par transformée de Fourier, la transformée en Z telle que

$$F(x^*(t)) = x(nh) \sum_{-\infty}^{+\infty} \exp(-jnh\omega) = x(nh) \sum_{-\infty}^{+\infty} Z^{-n}.$$

Un pôle de la fonction temps continu $X(p)$, transformée de Laplace de la fonction $x(t)$, est transformé par intégration numérique en un pôle "équivalent" en temps discret dans le plan Z , la position exacte de ce pôle étant fonction de la méthode utilisée.

5.4.1 Intégration implicite d'Euler

La méthode d'intégration implicite d'Euler sécrivant

$$x_{k+1} = x_k + h \dot{x}_{k+1}.$$

et l'opérateur dérivation de la transformée de Laplace étant la variable p , l'intégration numérique réalise la transformation

$$p = \frac{1 - Z^{-1}}{h}.$$

Un pôle quelconque $p_i = \sigma_i + j\omega_i$ du plan P est transformé en pôle

$$Z_i = \frac{1}{1 - (\sigma_i + j\omega_i)h} = \frac{(1 - \sigma_i h) + j\omega_i h}{(1 - \sigma_i h)^2 + (\omega_i h)^2} = X + jY.$$

La transformation d'un pôle p_i sur l'axe imaginaire ($\sigma_i = 0$) ne s'effectue pas sur le cercle unité ($|Z_i| \neq 1$) sauf pour $\omega_i = 0$. Les valeurs X et Y étant liées par la relation

$$(X - \frac{1}{2})^2 + Y^2 = (\frac{1}{2})^2$$

l'axe imaginaire du plan P est transformé en cercle de rayon $1/2$, centré sur l'abscisse $X = 1/2$ du plan Z . Tous les points à l'intérieur du cercle considéré sont l'image d'un point du demi-plan gauche de la variable P à partie réel négative (*figure 16*). La stabilité est préservée et ainsi, on redémontre que la méthode numérique est stable. L'axe imaginaire et une partie du demi-plan droit étant transformé à l'intérieur du cercle unité, la transformation amortit dans le plan Z un circuit éventuellement non amortie dans le plan d'origine P , elle a tendance à renforcer la stabilité. Cet artefact de simulation gênant pour l'étude des oscillateurs et des circuits à très haut coefficient de qualité ne peut être corrigé qu'en prenant un pas de calcul h très petit.

5.4.2 Intégration trapézoïdale

Avec

$$x_{k+1} = x_k + \frac{h}{2}(\dot{x}_{k+1} + \dot{x}_k),$$

l'intégration trapézoïdale réalise la transformation

$$p = \frac{2}{h} \frac{1 - Z^{-1}}{1 + Z^{-1}}.$$

Un pôle quelconque $p_i = \sigma_i + j\omega_i$ du plan P est transformé en pôle

$$Z_i = \frac{(\frac{2}{h} + \sigma_i) + j\omega_i}{(\frac{2}{h} - \sigma_i) - j\omega_i}.$$

L'axe imaginaire ($\sigma = 0$) du plan P est transformé intégralement en cercle unité dans le

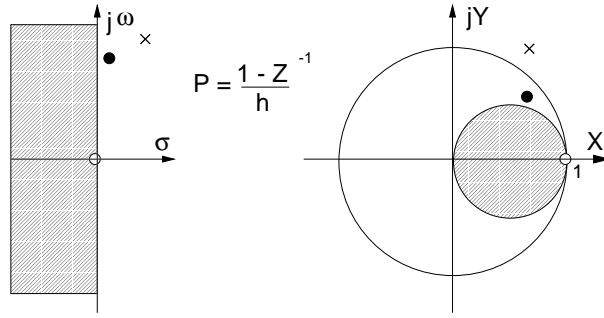


FIGURE 16 –

Traduction de la méthode d'intégration implicite d'Euler en terme de pôles dans le plan Z

plan Z et le demi-plan gauche du plan P correspond avec l'intérieur du cercle unité du plan Z montrant en outre que la stabilité est préservée (la méthode numérique est stable). Mais comme il été montré temporellement, l'intégration trapézoïdale peut générer de faibles oscillations parasites sur certains circuits.

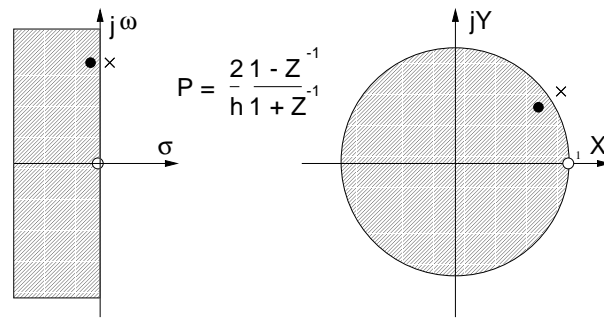


FIGURE 17 –

Traduction de la méthode d'intégration trapézoïdale en terme de pôles dans le plan Z

5.4.3 Intégration Gear2

Avec

$$x_{k+1} = \frac{4}{3}x_k - \frac{1}{3}x_{k-1} + \frac{2h}{3} \dot{x}_{k+1},$$

soit

$$p = \frac{1}{p} \left(\frac{1}{2}Z^{-2} - 2Z^{-1} + \frac{3}{2} \right),$$

un pôle quelconque $p_i = \sigma_i + j\omega_i$ du plan P est transformé en deux pôles

$$Z_i = \frac{2 \pm \sqrt{1 + 2h(\sigma_i + j\omega_i)}}{3 - 2h(\sigma_i + j\omega_i)}.$$

A chaque pôle du plan P correspond deux pôles dans le plan Z . Le demi-plan gauche étant entièrement transformé dans le plan Z à l'intérieur du cercle unité, la

méthode est stable. Cependant, l'axe imaginaire et une partie du demi-plan droit étant transformé à l'intérieur du cercle unité, la transformation amortit dans le plan Z un circuit éventuellement non amorti dans le plan P , elle a tendance à renforcer la stabilité et un circuit instable dans le plan P peut éventuellement être stable dans le plan Z . D'autre part, une partie de l'axe réel temps continu pouvant générer des paires de pôles complexes conjugués certains circuits peuvent présenter des phénomènes de suroscillation [?].

5.5 Conditionnement des méthodes d'intégration

Les techniques de conditionnements principalement utilisés en simulation électrique afin d'adapter les performances des méthodes d'intégration au circuit à analyser, sont essentiellement basées sur le contrôle dynamique du pas d'intégration. Dans le principe, si les variables calculées sont temporellement stables le pas d'intégration est augmenté, dans le cas contraire, il est réduit.

5.5.1 Contrôle de la LTE

Ce type de conditionnement du pas d'intégration est effectuée à partir d'une mesure de la différence entre la LTE calculée sur le pas considéré et la LTE prédite par une extrapolation polynomiale (typiquement une parabole) sur les échantillons précédents. En cas de dérive de la LTE, le pas d'intégration est automatiquement adapté.

5.5.2 Contrôle du pas d'intégration avec commutation d'algorithme

Les erreurs d'intégration sont d'autant plus grandes que les variations sur les signaux sont importantes. Ces variations ont souvent pour origine les transitions sur les différentes sources indépendantes, elles sont donc prévisibles et les erreurs peuvent ainsi être éventuellement contrôlées par un réglage du pas d'intégration et éventuellement une commutation d'algorithme sur Euler implicite qui est reconnue pour être une méthode d'intégration supportant relativement bien les fortes variations. Pour ce faire, on fait coïncider le début du pas d'intégration avec chaque transition détectée en diminuant automatiquement le pas d'intégration et éventuellement on commute l'algorithme.

5.5.3 Contrôle du pas d'intégration par comptage du nombre d'itérations

Les méthodes d'intégration sont uniquement utilisées en analyse transitoire et sont en boucle avec une méthode itérative d'analyse non linéaire (*figure 3*) sujette à des problèmes de divergence d'algorithme. Une amélioration notable peut être obtenue en réduisant le pas d'intégration en cas de détection de problème de convergence. D'autre part, dans le cas contraire, un comptage du nombre d'itérations étant effectué, si ce nombre est inférieur à une valeur prédéfinie (quelques unités), le pas d'intégration est augmenté. Toutefois, ce pas ne peut jamais être supérieur à une valeur fixée par l'utilisateur qui a ainsi, un contrôle sur la précision de l'analyse.

5.5.4 Adaptation de Gear2

Les formules de Gear ayant été déterminées pour un pas d'intégration constant, elles ne sont plus valables en cas de pas d'intégration variables. Ainsi, pour être utilisable, la

formule de Gear2 a été modifiée [?] et prend la forme

$$\dot{x}_{k+1} = \frac{2h_k + h_{k-1}}{h_k(h_k + h_{k-1})}x_{k+1} + \frac{h_k + h_{k-1}}{h_k h_{k-1}}x_k + \frac{h_k}{h_{k-1}(h_k + h_{k-1})}x_{k-1}.$$

Certaines rares combinaisons de h_k et h_{k-1} peuvent conduire à une instabilité de la méthode.

5.6 Choix de la variable d'intégration

Lors de la simulation en transitoire de certains dispositifs comportant des capacités non-linéaires (typiquement comportant des transistors MOS), il a été observé sur certains simulateurs électriques (SPICE,...) des imprécisions de calculs généralement attribués au "problème de conservation de la charge" [?], pouvant avoir deux sources totalement différentes. Si la première source d'erreur est indépendante du simulateur puisqu'elle a été clairement identifiée comme étant un problème de [modélisation](#)

qui peut être résolu par une meilleure représentation des phénomènes capacitifs décrivant le fonctionnement des transistors [?] [?], la seconde source est dépendante du simulateur puisqu'elle a pour origine le choix de la variable d'intégration (variable d'état) utilisée par la méthode d'intégration. Ceci est illustré sur l'exemple de la *figure 18* [?].

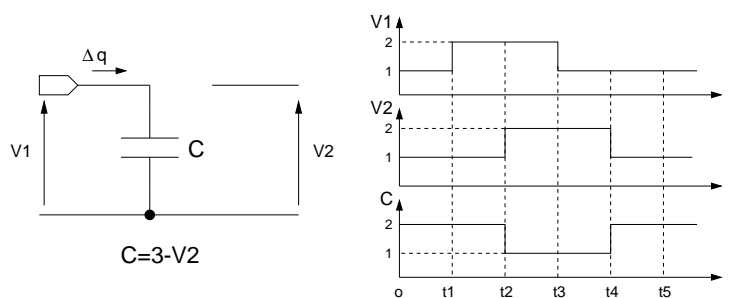


FIGURE 18 –

Illustration du problème de conservation de la charge

Pour ce faire, on considère un circuit (irréaliste) constitué d'un condensateur non-linéaire $C = f(V2) = 3 - V2$. A la fin du cycle de charge et de décharge, le circuit revenant dans son état d'origine, doit être caractérisé par un transfert de charge globalement nul. Toutefois, si on établit le bilan des transferts en écrivant

$$\begin{aligned} \Delta Q(t1) &= C (V1(t1) - V1(t0)) = 2 c \\ \Delta Q(t2) &= C (V1(t2) - V1(t1)) = 0 c \\ \Delta Q(t3) &= C (V1(t3) - V1(t2)) = -1 c \\ \Delta Q(t4) &= C (V1(t4) - V1(t3)) = 0 c, \end{aligned}$$

ce qui correspond a un transfert global de 1 Coulomb, on viole manifestement le principe de conservation de charge. Cette violation du principe fondamental de l'électrostatique, provient d'une écriture imparfaite des différents transferts de charge en partant de la relation simplifiée (un simulateur ne manipule que la dérivée des tensions ou courants inconnus)

$$i = C \frac{dV}{dt} = \frac{dQ}{dt}$$

au lieu la relation légitime

$$i = \frac{dQ}{dt} = C \frac{dV}{dt} + V \frac{dC}{dt}.$$

En effet, l'utilisation de cette dernière expression conduit à une variation de charge de $-2c$ en t_2 (au lieu de 0) et de $1c$ en t_4 (au lieu de 0) impliquant un transfert globalement nul.

Différentes approches peuvent être utilisées dans les simulateurs pour conserver la charge.

- En considérant la charge comme variable d'intégration :

à partir d'une méthode d'intégration implicite par exemple Euler, en prenant la tension comme variable d'intégration avec

$$V(n+1) = V(n) + h \dot{V}(n+1) \text{ et } i = C \dot{V}$$

on écrit

$$i(n+1) = \frac{C(n+1)}{h} (V(n+1) - V(n)),$$

alors qu'en prenant la charge comme variable d'intégration avec

$$Q(n+1) = Q(n) + h \dot{Q}(n+1) \text{ et } i = \dot{Q}$$

on écrit

$$i(n+1) = \frac{1}{h} (Q(n+1) - Q(n)).$$

On s'affranchit donc totalement de la valeur des condensateurs qui n'apparaît pas explicitement dans l'expression du courant, bien entendu toutes les relations analytiques modélisant le comportement des condensateurs non-linéaires devront être écrites sous formes d'équations de charge.

- En associant à un condensateur non-linéaire un équivalent hybride :

avec

$$C = f(V_1, V_2, V_3 \dots), \quad dC = \frac{\partial C}{\partial V_1} dV_1 + \frac{\partial C}{\partial V_2} dV_2 + \dots$$

et

$$i = C \frac{dV_1}{dt} + V_1 \frac{dC}{dt}$$

soit

$$i = C \frac{dV_1}{dt} + V_1 \frac{\partial C}{\partial V_1} \frac{dV_1}{dt} + V_1 \frac{\partial C}{\partial V_2} \frac{dV_2}{dt} + V_1 \frac{\partial C}{\partial V_3} \frac{dV_3}{dt} + \dots$$

on peut directement modéliser un condensateur par le schéma hybride équivalent correspondant de la *figure 19* [?]

- En associant à un condensateur non-linéaire un équivalent aux transcapacitances :

avec

$$Q = f(V_1, V_2, V_3 \dots) \text{ et } dQ = \frac{\partial Q}{\partial V_1} dV_1 + \frac{\partial Q}{\partial V_2} dV_2 + \frac{\partial Q}{\partial V_3} dV_3 + \dots$$

et

$$i = \frac{dQ}{dt} = \frac{\partial Q}{\partial V_1} \frac{dV_1}{dt} + \frac{\partial Q}{\partial V_2} \frac{dV_2}{dt} + \frac{\partial Q}{\partial V_3} \frac{dV_3}{dt} + \dots$$

soit

$$i = C \frac{dV_1}{dt} + C m_2 \frac{dV_2}{dt} + C m_3 \frac{dV_3}{dt} + \dots$$

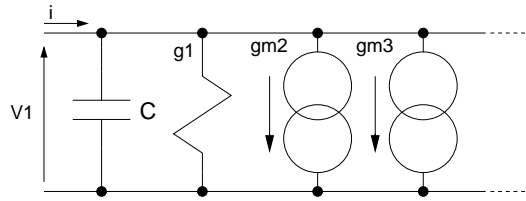


FIGURE 19 –

Une première modélisation prenant en compte la conservation de la charge

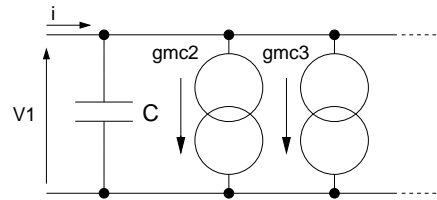


FIGURE 20 –

Une seconde modélisation prenant en compte la conservation de la charge

on peut directement modéliser un condensateur par le schéma équivalent aux transcapacitances correspondant de la *figure 20*

Malheureusement certains simulateurs, indépendamment du fait qu'il peuvent utiliser des modèles capacitifs exprimés directement en charge, utilisent comme relation de branche capacitive l'expression simplifiée, de telle sorte que le calcul numérique de la charge stockée sur un nœud capacitif à partir de l'intégration d'une tension fait intervenir la valeur du condensateur uniquement considérée aux bornes de l'intégrale, ne dépendant donc que de sa valeur finale et sa valeur initiale, générant ainsi une erreur de troncature. Dans ce cas, le seul paramètre dont dispose l'utilisateur pour obtenir la précision désirée est d'imposer un pas de calcul très petit afin de "moyenner" les erreurs de transfert de charge. Ceci peut être facilement vérifié sur l'exemple de la *figure 18*. On notera que les mêmes causes produisant les mêmes effets, un mauvais réglage des paramètres de conditionnement de l'analyse transitoire peut produire également des erreurs de troncature induisant une perte de conservation de la charge.

5.7 Choix d'une méthode d'intégration

Pratiquement, du fait du conditionnement des méthodes, il n'est pas facile de faire à priori le choix d'une méthode d'intégration. Si le choix entre les méthodes se fonde essentiellement sur l'erreur de troncature locale la méthode de Gear est meilleure que la méthode trapézoïdale qui est meilleure que la méthode d'Euler implicite. En terme de stabilité les trois méthodes sont équivalentes. Le choix se situent pratiquement au niveau des artefacts de simulation, suramortissement pour Euler et Gear et propension à donner des oscillations parasites pour trapézoïdale qui feront choisir une méthode plutôt qu'une autre pour un circuit particulier afin de ne pas avoir à adopter un pas d'intégration trop petit pour des circuits fortement relachés. En règle générale pour les circuits comportant beaucoup de transitions, comme les circuits analogiques temps discret utilisant la technique des capacités commutées, l'intégration par Euler donne de

bons résultats, compte tenu des nombreuses commutations de signaux.

6 Méthodes numériques d'analyse non linéaires

Si un circuit contient des composants non-linéaires, il requiert pour l'analyse statique et l'analyse transitoire, une méthode numérique pour résoudre une équation algébrique non linéaire implicite de la forme

$$G(\mathbf{x}) = 0.$$

Pour l'analyse statique, la forme précédente est respectée par principe, puisque toutes les dérivées sont nulles et pour l'analyse transitoire, la forme précédente est respectée à l'instant k , de part l'utilisation d'une méthode d'intégration numérique. Pratiquement, en simulation électrique les méthodes d'analyse utilisées sont basées sur la méthode de Newton-Raphson.

6.1 Le principe de l'algorithme de Newton-Raphson

Une méthode de Newton-Raphson d'analyse non linéaire détermine itérativement la solution de $g(x) = 0$ sous la condition que la dérivée existe. Ainsi, partant d'une solution $x^{(0)}$ pas trop "éloignée" de la vraie solution \bar{x} , on détermine une meilleur approximation

$$\begin{aligned} x^{(1)} &= x^{(0)} - \frac{g(x^{(0)})}{g'(x^{(0)})} \\ &\vdots \\ x^{(n)} &= x^{(n-1)} - \frac{g(x^{(n-1)})}{g'(x^{(n-1)})} \\ x^{(n+1)} &= x^{(n)} - \frac{g(x^{(n)})}{g'(x^{(n)})} \end{aligned}$$

jusqu'à ce que la différence entre les deux dernières valeurs calculées soit négligeable. Cet algorithme a une interprétation géométrique simple puisque en écrivant la n^{ieme} équation précédente sous la forme

$$g(x^{(n)}) = g'(x^{(n)})(x^{(n)} - x^{(n+1)}),$$

on voit sur la *figure 21* que c'est l'équation d'une droite de pente $g'(x^{(n)})$ interceptant l'axe des x au point $x^{(n+1)}$. Ainsi, à chaque itération, on construit une droite de pente $g'(x^{(n)})$ à partir du point $(x^{(n)}, g(x^{(n)}))$, l'intersection avec l'axe des x étant l'approximation suivante.

En terme de vitesse de convergence, l'algorithme de Newton-Raphson est quadratique, puisque l'erreur au n^{ieme} pas est proportionnelle au carré de l'erreur du pas précédent. En effet, avec

$$\begin{aligned} \epsilon^{(n)} &= \bar{x} - x^{(n)} \\ \epsilon^{(n+1)} &= \bar{x} - x^{(n+1)} \\ \epsilon^{(n+1)} &= \bar{x} - x^{(n)} + \frac{g(x^{(n)})}{g'(x^{(n)})} \\ \epsilon^{(n+1)} &= \epsilon^{(n)} + \frac{g(x^{(n)})}{g'(x^{(n)})} \end{aligned}$$

et à partir du développement en série de Taylor de $g(x)$ au voisinage de sa racine \bar{x}

$$g(x) = g(\bar{x}) + g'(\bar{x})(x - \bar{x}) + \frac{1}{2}g''(\bar{x})(x - \bar{x})^2 + \dots,$$

avec

$$g(\bar{x}) = 0$$

et

$$g(x^n) = -\epsilon^{(n)}g'(\bar{x}) + \frac{(\epsilon^{(n)})^2}{2}g''(\bar{x}) + \dots,$$

divisée par $g'(\bar{x})$ et substituée dans $\epsilon^{(n+1)}$, on détermine

$$\epsilon^{(n+1)} = \frac{(\epsilon^{(n)})^2}{2} \frac{g''(\bar{x})}{g'(\bar{x})}.$$

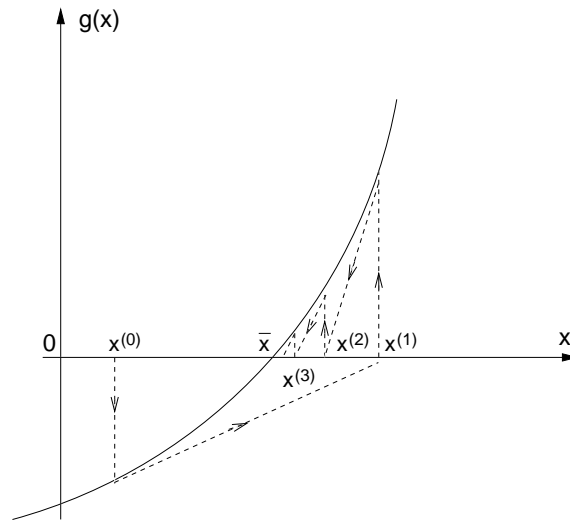


FIGURE 21 –

Interprétation géométrique de l'algorithme de Newton-Raphson

6.2 L'algorithme de Newton-Raphson pour un système d'équations

Si la fonction $G(\mathbf{x})$ est dérivable au voisinage de $\mathbf{x} = \vec{x}^{(n)} = [x_1^{(n)}, \dots, x_m^{(n)}]^T$, elle a pour série de Taylor

$$G(\mathbf{x}) = G(\mathbf{x}^{(n)}) + \mathbf{J}(\mathbf{x}^{(n)})(\mathbf{x} - \mathbf{x}^{(n)}) + \dots$$

avec $G(\mathbf{x}) = [g_1(x), \dots, g_m(x)]^T$ et $\mathbf{J}(\mathbf{x}^{(n)})$ étant le Jacobien (matrice de Jacobi) tel que

$$\mathbf{J}(\mathbf{x}^{(n)}) = \begin{pmatrix} \frac{\partial g_1}{\partial x_1} \Big|_{x^{(n)}} & \frac{\partial g_1}{\partial x_2} \Big|_{x^{(n)}} & \cdots & \frac{\partial g_1}{\partial x_m} \Big|_{x^{(n)}} \\ \frac{\partial g_2}{\partial x_1} \Big|_{x^{(n)}} & \frac{\partial g_2}{\partial x_2} \Big|_{x^{(n)}} & \cdots & \frac{\partial g_2}{\partial x_m} \Big|_{x^{(n)}} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial g_m}{\partial x_1} \Big|_{x^{(n)}} & \frac{\partial g_m}{\partial x_2} \Big|_{x^{(n)}} & \cdots & \frac{\partial g_m}{\partial x_m} \Big|_{x^{(n)}} \end{pmatrix}.$$

Sur l'itération suivante, en négligeant les termes d'ordre supérieur à 2 du développement,

on peut écrire

$$G(\mathbf{x}^{(n+1)}) = G(\mathbf{x}^{(n)}) + \mathbf{J}(\mathbf{x}^{(n)})(\mathbf{x}^{(n+1)} - \mathbf{x}^{(n)}),$$

et si $\mathbf{x}^{(n+1)}$ est une solution suffisamment proche de la vraie solution $\bar{\mathbf{x}}$, on obtient avec $G(\mathbf{x}^{(n+1)}) = 0$

$$G(\mathbf{x}^{(n)}) + \mathbf{J}(\mathbf{x}^{(n)})(\mathbf{x}^{(n+1)} - \mathbf{x}^{(n)}) = 0,$$

ainsi que la formule dite "de l'algorithme de Newton-Raphson"

$$\mathbf{x}^{(n+1)} = \mathbf{x}^{(n)} - [\mathbf{J}(\mathbf{x}^{(n)})]^{-1}G(\mathbf{x}^{(n)}).$$

On notera qu'on peut éviter l'inversion du Jacobien, en résolvant sur chaque itération directement l'équation linéarisée de Newton-Raphson

$$\mathbf{J}(\mathbf{x}^{(n)}) \mathbf{x}^{(n+1)} = \mathbf{J}(\mathbf{x}^{(n)}) \mathbf{x}^{(n)} - G(\mathbf{x}^{(n)})$$

par une *méthode numérique de résolution d'un système d'équations linéaires*.

6.3 Circuit équivalent temps discret de Newton-Raphson

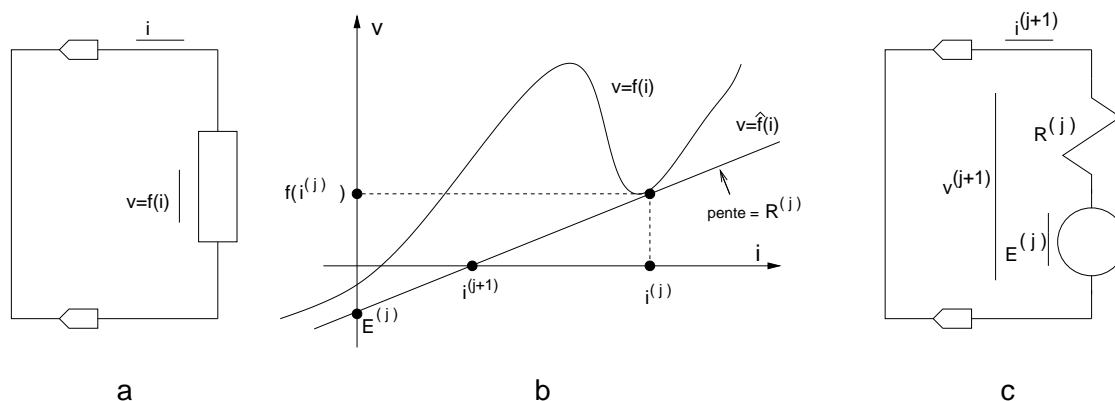


FIGURE 22 –

Illustration de l'équivalence temps discret de Newton-Raphson

Considérons un circuit caractérisé par une équation de constitution en courant et simplement constitué d'un dipôle non-linéaire (*figure 22 a*), si le dipôle est court-circuité, il est décrit par l'équation

$$f(i) = 0.$$

Résolvons cette équation par la méthode de Newton-Raphson. Partant d'une valeur initiale $i^{(0)}$, l'application de la formule de l'algorithme nous donne après j itérations

$$i^{(j+1)} = i^{(j)} - (\mathcal{R}^{(j)})^{-1} f(i^{(j)})$$

avec par définition

$$\mathcal{R}^{(j)} = \left. \frac{\partial f(i)}{\partial i} \right|_{i=i^{(j)}}.$$

Supposons maintenant qu'à la j^{ieme} itération on remplace la fonction non-linéaire $v = f(i)$ par la fonction affine (*figure 22* b et c)

$$\hat{f}(i) = v = \mathcal{R}^{(j)}i + E^{(j)},$$

et qu'on applique la formule de l'algorithme de Newton-Raphson. A la j^{ieme} itération, on obtient la relation

$$i^{(j+1)} = i^{(j)} - (\mathcal{R}^{(j)})^{-1} \hat{f}(i^{(j)})$$

soit

$$i^{(j+1)} = i^{(j)} - (\mathcal{R}^{(j)})^{-1} f(i^{(j)})$$

puisque pour $i = i^{(j)}$ on a $\hat{f}(i^{(j)}) = f(i^{(j)})$ (*figure 22* b). Ainsi, à la j^{ieme} itération et avec

$$E^{(j)} = f(i^{(j)}) - \mathcal{R}^{(j)}i^{(j)},$$

les circuits a et c de la *figure 22* sont électriquement équivalents puisque fournissant la même réponse. Pour une branche caractérisée par une équation de constitution en tension on a le résultat dual de la *figure 23* avec

$$\mathcal{G}^{(j)} = \left. \frac{\partial f(v)}{\partial v} \right|_{v=v^{(j)}} \quad \text{et} \quad I^{(j)} = f(v^{(j)}) - \mathcal{G}^{(j)}v^{(j)}$$

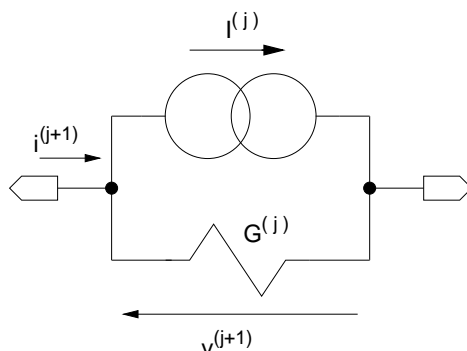


FIGURE 23 –

Equivalent temps discret de Newton-Raphson d'une branche définie en tension

En généralisant, on peut dire [?] que résoudre un circuit non linéaire par la méthode de Newton-Raphson, revient à résoudre un circuit résistif linéaire équivalent temps discret associé sur chaque itération. Le circuit équivalent restant topologiquement identique à chaque itération, il suffit d'effectuer une actualisation des valeurs des composants.

Ainsi, l'application directe du formalisme de la méthode de Newton-Raphson conduisant à une évaluation de m^2 dérivées sur chaque itération, numériquement parlant, il est plus efficace de linéariser individuellement chaque branche du réseau par l'équivalence temps discret de Newton-Raphson et de formuler directement le jeu d'équations ainsi obtenu.

6.4 Critères de convergence de l'algorithme

Comme toutes les procédures itératives, la méthode de Newton-Raphson a besoin d'un critère pour déterminer si la solution a convergé. Pour ce faire, on peut mesurer

la différence entre les deux dernières valeurs calculées et mettre fin à l'itération lorsque cette différence est négligeable. En terme mathématique chaque composante du vecteur $x_i^{(n+1)}$ doit satisfaire le critère dit "de la différence"

$$|x_i^{(n+1)} - x_i^{(n)}| < \epsilon_x.$$

En simulation électrique cette forme simple de critère de convergence pose un certain nombre de problèmes. Ainsi, si la variable considérée est une tension (ou un courant), elle ne garantit pas que les lois de Kirchhoff en courant (ou tension) soient obligatoirement respectées. Supposons pour concrétiser que la variable x_i soit une tension. Si on considère un nœud basse impédance (par exemple sur une jonction PN fortement polarisée en direct), le critère de la différence peut être parfaitement respecté (d'autant plus que la convergence est lente) et la loi de Kirchhoff en courant très loin d'être respectée du fait de la forte amplitude des courants sur le nœud considéré. Pour palier à cette imperfection, la plupart des simulateurs électriques associe au critère de la différence un second critère dit "du résidu" tel que

$$|f_i(v^{(n+1)})| = \sum |i_i^{(n+1)}| < \epsilon_i.$$

Une autre modification doit être apportée si on considère que les critères utilisés ne doivent pas dépendre des amplitudes des variables manipulées et prendre en compte le fait que la précision demandée ne peut pas être inférieure à la précision intrinsèque du calculateur (l'algorithme bouclerait indéfiniment). Pratiquement, le critère de la différence s'écrit

$$|v_i^{(n+1)} - v_i^{(n)}| < \epsilon_a + \epsilon_r \max\{|v_i^{(n)}|, |v_i^{(n+1)}|\}.$$

Le paramètre de réglage ϵ_a est le critère de tolérance absolue (typiquement $\approx 1\mu V$) et ϵ_r est le critère de convergence relative (typiquement $\approx 0,001$). Et pour le résidu l'algorithme devra respecter la contrainte

$$|f_i(v^{(n+1)})| < \epsilon_a + \epsilon_r |i_{max}^{(n+1)}|.$$

Typiquement $\epsilon_a \approx 1pA$ et $\epsilon_r \approx 10^{-6}$.

6.5 Conditionnement de la méthode

Une divergence de la méthode de Newton-Raphson peut avoir essentiellement quatre causes. La première a pour origine la nature des lois mathématiques utilisées pour la modélisation des composants actifs, la seconde est due à la non-monotonie des non-linéarités, la troisième provient des propriétés du circuit qui peut par construction ne pas posséder de point de fonctionnement stable (oscillateur, astables, ...) ou en posséder plusieurs (bistables, bascules numériques, ...). Enfin, la quatrième, très incertaine et également générée par le [modèle](#)

est due à la distorsion de raccordement entre les différentes zones de fonctionnement des dispositifs actifs, faisant que le modèle n'est pas toujours dérivable.

Pour palier à la divergence, deux catégories de conditionnement peuvent éventuellement être utilisées en simulation électrique : le conditionnement par limitation simple et le conditionnement par le principe du continuum. Un troisième type de conditionnement qui n'aide pas à la convergence peut être utilisé en analyse transitoire et en statique multipoint (caractéristique de transfert), c'est le conditionnement par réduction d'erreur et "bypass" qui est essentiellement une technique d'accélération de la convergence.

6.5.1 Conditionnement par limitation simple

Bien que lors de l'implantation d'un modèle de composant actif dans un simulateur électrique, le modélisateur a pour devoir de limiter analytiquement l'amplitude des courants et des tensions pouvant être générés par le composant à des valeurs "physiquement" raisonnables, de part la nature des lois mathématiques utilisées, la différence entre deux valeurs de variable correspondant à deux itérations successives peut être suffisamment grande pour entraîner une divergence de l'algorithme de Newton-Raphson. Le conditionnement par limitation à pour fonction d'écrêter la valeur des incréments de tension ou de courant sur une valeur prédéfinie associée au composant actif considéré. Il existe de nombreuses techniques, ainsi pour une diode ayant une équation de constitution de la forme

$$Id = Is(e^{\frac{Vd}{Vt}} - 1),$$

on peut limiter arbitrairement l'incrément de tension à une valeur n'excédant pas quelques V_t ($\approx 26mV$), la fonction inverse $Vd = f(Id)$ étant connue, on peut également selon la valeur du courant calculé sur la précédente itération, opter pour une incrémentation en courant,

6.5.2 Conditionnement par le principe du continuum

Si nous considérons le circuit parallèle de la *figure 24* constitué d'une résistance et d'une diode tunnel polarisée par une source de courant statique. La diode ayant pour caractéristique la fonction non-linéaire $Id = f(Vd)$, il est clair que le circuit possède un point de fonctionnement stable en C puisque par les lois de Kirchhoff, le réseau est contraint de respecter la relation

$$Id = f(Vd) = I_0 - Vd/R.$$

Cependant on peut constater graphiquement que l'application de méthode de Newton-Raphson ne convergera pas si la valeur initiale est V_1 puisque dans ce cas l'algorithme oscillera indéfiniment entre V_1 et V_2 . Sur cet exemple simple particulier, la divergence est due à la non-monotonie de la non-linéarité de la diode tunnel, mais pratiquement, ce type de non-monotonie peut apparaître sur des circuits plus compliqués par combinaison de plusieurs non-linéarités monotones.

Pour sortir de cet état de divergence, le simulateur électrique peut utiliser le principe dit "du continuum". Pour ce faire, un paramètre du circuit est modifié ou ajouté pour obtenir une topologie connue pour être facilement simulable, et une fois que la convergence est constatée le circuit retourne graduellement à sa forme d'origine, sachant que sur chaque modification du paramètre l'algorithme est initialisé avec des valeurs proches de leurs valeurs finales. La convergence est ainsi améliorée, mais sans être garantie, et avec des temps de simulation très fortement allongés.

- La technique de la source graduelle :

Une première analyse est réalisée avec toutes les sources indépendantes de tension et de courant nulles. Puis les tensions et les courants sont restitués graduellement et une analyse est réalisée sur chaque changement avec pour valeurs initiales les valeurs précédemment calculées, jusqu'à ce que les sources de tension et de courant soient restituées dans leur état d'origine.

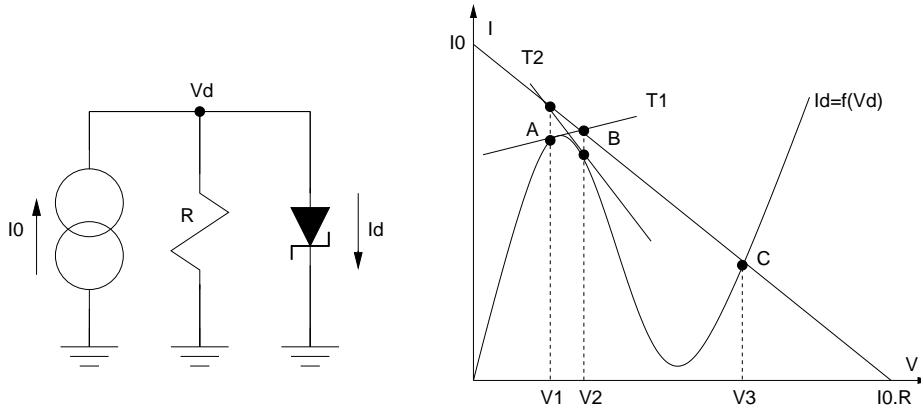


FIGURE 24 –

Circuit diode tunnel et résistance

- La technique du gmin graduelle :

Dans une première technique, l'analyse est initialisée avec une faible valeur de résistance placée en parallèle sur tous les dispositifs non-linéaires. La solution est normalement facilement calculée puisque les non-linéarités sont physiquement amorties. Ensuite la valeur des résistances est graduellement augmentée pour atteindre une valeur pour laquelle elles peuvent être enlevées. La seconde technique consiste à partir d'une valeur gmin systématiquement placée sur tous les dispositifs non-linéaires et suffisamment petite pour être sans influence (typiquement 10^{-12}), et à augmenter cette valeur jusqu'à ce que la convergence soit atteinte.

- La technique du pseudotransitoire :

Un condensateur de forte valeur ($\approx 1F$) et de tension initiale nulle est placée en parallèle sur tous les dispositifs non-linéaires, et un transitoire (pseudotransitoire) partant de zéro est réalisé jusqu'à ce que une solution soit trouvée. Les condensateurs sont ensuite enlevés et les tensions et courant de repos conservés. Une technique très voisine consiste à placer une inductance de forte valeur ($\approx 1H$) en série avec les sources de tension. Dans tous les cas, on doit veiller à ce que le circuit ne soit pas transformé en oscillateur par l'adjonction des différents éléments parasites.

6.5.3 Conditionnement par réduction d'erreur et "bypass"

Lors d'une analyse transitoire ou une analyse statique multipoint, il est raisonnable de penser que les valeurs des tensions et des courants obtenues lors de la précédente analyse peuvent être une bonne estimation pour la solution à déterminer. Cette estimation peut être encore améliorée si une extrapolation est réalisée sur un jeu de valeurs constitué de plusieurs analyses précédentes. Cette technique de détermination des valeurs initiales peut éventuellement être utilisée avec profit afin d'accélérer la convergence de la procédure de Newton-Raphson.

Une seconde technique d'accélération peut également être utilisée, c'est la technique dite du "bypass" pour laquelle les relations de branches non-linéaires et les dérivées ne sont calculées que si les arguments de ces relations changent de manière "appréciable", sinon on conserve les valeurs de l'itération précédente. Si cette technique est intéressante dans

le principe, puisque un circuit numérique peut être inactif (latent) sur une grande période de temps, elle nécessite la mémorisation de toutes les variables précédentes (comme le conditionnement par réduction d'erreur), la prise de décision concernant l'inactivité du circuit n'est pas évidente et lors de la sortie de l'inactivité on peut avoir, selon la grandeur du pas de calcul, des sauts de valeurs générateurs de bruit et de problème de convergence pouvant dégrader les résultats de simulation [?].

6.5.4 Caractéristiques des simulateurs de troisième génération

L'objectif des simulateurs électriques de troisième génération est d'essayer de satisfaire aux contraintes à priori contradictoires : analyser des circuits de grande complexité (plusieurs milliers d'éléments non-linéaires) rapidement et avec la même précision que les simulateurs classiques. Pour ce faire, le partitionnement structurel [?], c'est à dire une décomposition du circuit à analyser en blocs élémentaires est largement utilisé. Sous certaines conditions, principalement si l'interaction entre blocs est négligeable, ceci permet d'exploiter le parallélisme, l'identité entre blocs, l'inactivité (la latence), éventuellement les méthodes d'intégration explicites, et surtout l'utilisation de techniques de résolutions matricielles essentiellement basées sur des techniques itératives telles que Gauss-Jacobi et Gauss-Seidel [?] [?] avec relaxation [?]. Le défaut principal de la plupart des simulateurs de troisième génération est qu'il est nécessaire de supposer à priori que les blocs élémentaires sont électriquement découplés et si à posteriori on constate qu'il ne le sont pas, il est nécessaire, soit de les réévaluer, dans ce cas les temps de calculs peuvent être du même ordre de grandeur que ceux des simulateurs standards ou de ne pas les réévaluer, dans ce cas des erreurs d'analyse sont inévitablement commises. De ce fait, pratiquement les simulateurs de troisième génération sont principalement utilisés en tant que simulateur temporel pour la circuiterie numérique.