

Présentation TIPE

Trouver l'équation différentielle d'un circuit électronique algorithmiquement.

Léo Kosman

MP2I — Lycée Thiers

Idée générale

On considère les circuits électroniques composés de dipôles (résistors, compdensateurs, bobines et générateurs idéaux de tension et d'intensité).

Idée générale

On considère les circuits électroniques composés de dipôles (résistors, condensateurs, bobines et générateurs idéaux de tension et d'intensité). On veut créer un algorithme qui peut

trouver une équation différentielle vérifiée par une grandeur du circuit. Et si l'on peut choisir la grandeur arbitrairement, c'est encore mieux !

Idée générale

On considère les circuits électroniques composés de dipôles (résistors, compdensateurs, bobines et générateurs idéaux de tension et d'intensité). On veut créer un algorithme qui peut

trouver une équation différentielle vérifiée par une grandeur du circuit. Et si l'on peut choisir la grandeur arbitrairement, c'est encore mieux !

Note : on veut trouver l'équation différentielle littéralement, et pas résoudre le circuit numériquement.

Quelques résultats obtenus

On peut résoudre certains circuits (assez simples) avec la méthode dite de "proche en proche". Cela consiste à faire des substitutions répétées.

Quelques résultats obtenus

On peut résoudre certains circuits (assez simples) avec la méthode dite de "proche en proche". Cela consiste à faire des substitutions répétées.

L'algorithme termine et est correct à condition de toujours trouver une substitution à chaque étape.

Quelques résultats obtenus

On peut résoudre certains circuits (assez simples) avec la méthode dite de "proche en proche". Cela consiste à faire des substitutions répétées.

L'algorithme termine et est correct à condition de toujours trouver une substitution à chaque étape.

Ses avantages : il permet de choisir la grandeur qu'on veut, et est relativement efficace ($O(n^2)$ a priori, si bien implémenté)

Quelques résultats obtenus

On peut résoudre certains circuits (assez simples) avec la méthode dite de "proche en proche". Cela consiste à faire des substitutions répétées.

L'algorithme termine et est correct à condition de toujours trouver une substitution à chaque étape.

Ses avantages : il permet de choisir la grandeur qu'on veut, et est relativement efficace ($O(n^2)$ a priori, si bien implémenté)

J'ai une implémentation en OCaml, mais elle n'est pas très performante, car je n'ai pas spécialement réfléchi à des structures de données plus adaptées dans certains cas.

Pistes de recherche et objectifs

- ▶ Implémenter les associations en série/parallèle des circuits pour trouver une équation différentielle

Pistes de recherche et objectifs

- ▶ Implémenter les associations en série/parallèle des circuits pour trouver une équation différentielle
- ▶ Faire une variante du proche en proche sans choix de la grandeur, mais qui permettrait de garantir une équation différentielle ?

Pistes de recherche et objectifs

- ▶ Implémenter les associations en série/parallèle des circuits pour trouver une équation différentielle
- ▶ Faire une variante du proche en proche sans choix de la grandeur, mais qui permettrait de garantir une équation différentielle ?
- ▶ Essayer de trouver un algorithme utilisant le théorème de Millman

Pistes de recherche et objectifs

- ▶ Implémenter les associations en série/parallèle des circuits pour trouver une équation différentielle
- ▶ Faire une variante du proche en proche sans choix de la grandeur, mais qui permettrait de garantir une équation différentielle ?
- ▶ Essayer de trouver un algorithme utilisant le théorème de Millman
- ▶ Optimiser l'implémentation du proche en proche en choisissant des structures de données plus adaptées.