

Sukzessive Index-Reduktion elektrischer Netzwerke mit Hilfe einer minimalen Anzahl von Zusatzelementen

Klaus Röbenack and Gunther Reißig

Abstract

43rd International Scientific Colloquium, Technical University of Ilmenau, September 21-24,
1998.

K. Röbenack / G. Reißig

Sukzessive Index-Reduktion elektrischer Netzwerke mit Hilfe einer minimalen Anzahl von Zusatzelementen

EINLEITUNG

Viele technische Systeme — insbesondere elektrische Netzwerke — lassen sich durch *Algebroadifferential-Gleichungen* (*Differential Algebraic Equations, DAEs*) beschreiben. Die *Regularität* und der *Index* sind von entscheidender Bedeutung für die analytischen und numerischen Eigenschaften einer DAE [1–3]. Gerade bei DAEs mit höherem Index, d. h. einem Index größer als eins, können kleine Störungen am Eingang beliebig große Fehler in der Lösung verursachen [3]. Eine Möglichkeit, diese Probleme bei der Schaltungssimulation zu vermeiden, besteht in der gezielten Modifikation des elektrischen Netzwerkes. In diesem Beitrag sind folgende Modifikationen des Netzwerkes erlaubt:

- (M1) Einfügen eines Widerstandes, einer Induktivität oder einer Kapazität in Reihe zu einem bereits vorhandenen Netzwerkelement
- (M2) Einfügen eines Widerstandes, einer Induktivität oder einer Kapazität zwischen zwei Knoten des Netzwerkes

Mit diesen Modifikationen lassen sich reguläre Netzwerke vom Index 1 generieren [4]. In der vorliegenden Arbeit wird ein Algorithmus zur Index-1-Regularisierung vorgestellt, der mit einer minimalen Anzahl von zusätzlichen Netzwerkelementen auskommt. Dabei wird keine Regularität des ursprünglichen Netzwerkes vorausgesetzt. Der Algorithmus basiert auf dem Konzept eines normalen Paares konjugierter Bäume.

THEORETISCHE GRUNDLAGEN

Betrachtet werden lineare Netzwerke, die Widerstände, Kapazitäten, Induktivitäten und unabhängige Quellen enthalten können und außerdem einen zusammenhängenden Netzwerkgraphen besitzen. Operationsverstärker werden durch Nullator-Norator-Paare modelliert. Gegeben sei ein solches Netzwerk \mathcal{N} mit b Zweigen und dem Parametervektor $p \in \mathbb{R}^k$, der alle Parameter von \mathcal{N} enthält: Widerstands-, Kapazitäts- und Induktivitätswerte. Das *Zweigspannungs-Zweigstrom-Gleichungssystem* von \mathcal{N} hat die Form

$$E(p) \dot{x}(t) = A(p) x(t) + f(t) \quad (1)$$

mit den $2b \times 2b$ -dimensionalen parameterabhängigen Matrizen E und A und dem Vektor x der Zweigspannungen und Zweigströme. Zur Untersuchung des Gleichungssystems (1) betrachtet man die Matrizenschar $sE - A$. Eine Matrizenschar heißt *regulär*, falls $\det(sE - A) \neq 0$. Andernfalls heißt die Matrizenschar *singulär*.

Jede reguläre Matrizenschar $sE - A$ kann in eine *Weierstraß-Normalform*

$$P(sE - A)Q = \begin{pmatrix} sI - W & 0 \\ 0 & sN - I \end{pmatrix}$$

transformiert werden [5, 6]. Dabei sind die Matrizen P und Q reguläre Transformationsmatrizen und I symbolisiert die Einheitsmatrix. Die für DAEs charakteristischen Eigenschaften werden durch die nilpotente Blockdiagonalmatrix $N = \text{diag}(N_1, \dots, N_d)$ bestimmt. Sie besteht aus Jordan-Blöcken der Form

$$N_i = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & & & 0 & 1 \\ 0 & 0 & \cdots & 0 & 0 \end{pmatrix}.$$

Der *Index* ist die Spaltenanzahl des größten Jordan-Blockes von N . Schreibweise: $\text{ind}(E, A)$.

Zu beachten ist insbesondere die Parameterabhängigkeit der Koeffizientenmatrizen $E(p)$ und $A(p)$ des Gleichungssystems (1). Ein Netzwerk \mathcal{N} heißt *generisch regulär* bzw. *generisch lösbar*, falls die zugehörige Matrizenchar $sE(p) - A(p)$ des Gleichungssystems (1) für alle p in einer offenen und dichten Teilmenge von \mathbb{R}^k regulär ist. Man sagt, ein generisch reguläres Netzwerk \mathcal{N} hat *generisch den Index* ν , wenn $\text{ind}(E(p), A(p)) = \nu$ für alle p einer offenen und dichten Teilmenge von \mathbb{R}^k gilt.¹

Zur Untersuchung generischer Eigenschaften linearer Netzwerke hat sich das Konzept des Paares konjugierter Bäume bewährt [7, 8]. Ein Paar aufspannender Bäume (t_1, t_2) des ungerichteten Netzwerkgraphen $\mathcal{G}_{\mathcal{N}}$ eines Netzwerkes \mathcal{N} heißt *Paar konjugierter Bäume*, wenn

- (i) der Baum t_1 alle Noratorzweige, keinen Nullatorzweig, alle Zweige mit Spannungsquellen, keinen Zweig mit Stromquellen und ggf. einige der Widerstands-, Induktivitäts- bzw. Kapazitätzweige enthält und
- (ii) der Baum t_2 alle Nullatorzweige, keinen Noratorzweig, alle Zweige mit Spannungsquellen, keinen Zweig mit Stromquellen und die gleichen Widerstands-, Induktivitäts- bzw. Kapazitätzweige wie t_1 enthält.

Ein Paar konjugierter Bäume heißt *normal*, wenn es unter allen Paaren konjugierter Bäume von \mathcal{N} die Summe aus der Anzahl der Kapazitäten in den Bäumen und der Anzahl der Induktivitäten, die nicht in den Bäumen enthalten sind, maximiert. Es bestehen folgende Zusammenhänge:

- (i) Ein Netzwerk \mathcal{N} ist genau dann generisch regulär, wenn $\mathcal{G}_{\mathcal{N}}$ ein Paar konjugierter Bäume besitzt.
- (ii) Ein Netzwerk \mathcal{N} hat genau dann den generischen Index 1, wenn $\mathcal{G}_{\mathcal{N}}$ ein Paar konjugierter Bäume besitzt, welches alle Kapazitätzweige und keinen Induktivitätszweig enthält.

INDEX-REDUKTION

Sei \mathcal{N} ein lineares Netzwerk, das die o. g. Voraussetzungen erfüllt und keine Masche aus Nullator- oder Noratorzweigen enthält. Dann liefert der in Tabelle 1 angegebene Algorithmus ein generisch reguläres Index-1-Netzwerk \mathcal{N}'' . Außerdem gibt es kein Netzwerk, das aus \mathcal{N} durch Anwendung der erlaubten Modifikationen (M1) und (M2) hervorgeht, generisch regulär sowie vom Index 1 ist und weniger Zusatzelemente benötigt (siehe [9, Lemma 1 und Theorem 1]).

Die Schritte 1 und 3 des Algorithmus können bis zu $\mathcal{O}(b^2)$ Netzwerkmodifikationen erfordern. Ein normales Paar konjugierter Bäume läßt sich z. B. mit der in [10] vorgeschlagen Methode bestimmen. Die Komplexität dieser Methode beträgt $\mathcal{O}(b^4)$. Da das Netzwerk \mathcal{N}' wegen

¹Die Matrizencharakteren anderer Netzwerkgleichungen, wie sie z. B. bei Knotenspannungs- oder Maschenstromanalyse anfallen, können beim gleichen Netzwerk zu anderen Aussagen hinsichtlich Regularität und Index führen.

Eingabe:	Netzwerk \mathcal{N} .
Schritt 1:	Konstruktion eines Netzwerkes \mathcal{N}' aus \mathcal{N} : <ul style="list-style-type: none"> (i) Ersetze jede Spannungsquelle durch eine Kapazität. (ii) Ersetze jede Stromquelle durch eine Induktivität. (iii) Erweitere \mathcal{N} um eine zusätzliche Induktivität zwischen allen Knotenpaaren, die noch nicht durch ein zweipoliges Netzwerkelement miteinander verbunden sind.
Schritt 2:	Bestimme ein normales Paar konjugierter Bäume (t_1, t_2) von \mathcal{N}' .
Schritt 3:	Konstruktion eines Netzwerkes \mathcal{N}'' aus dem ursprünglichen Netzwerk \mathcal{N} : <ul style="list-style-type: none"> (i) Einfügen eines Widerstandes oder einer Induktivität in Reihe zu jeder Kapazität und jeder Spannungsquelle aus \mathcal{N}, die nicht in t_1 enthalten ist. (ii) Einfügen eines Widerstandes oder einer Kapazität parallel zu jeder Induktivität und jeder Stromquelle aus \mathcal{N}, die in t_1 enthalten ist. (iii) Einfügen eines Widerstandes oder einer Kapazität zwischen jedem Knotenpaar, welches durch eine in t_1 und \mathcal{N}', aber nicht in \mathcal{N} enthaltene Induktivität verbunden ist.
Ausgabe:	Netzwerk \mathcal{N}'' .

Tabelle 1: Algorithmus zur Index-1-Regularisierung mit minimaler Anzahl von Zusatzelementen

Schritt 1(iii) bis zu $\mathcal{O}(b^2)$ Zweige haben kann, liegt der Rechenaufwand des gesamten Algorithmus bei $\mathcal{O}(b^8)$.

Die Anzahl der durchzuführenden Operationen ist damit zwar polynomial hinsichtlich der Anzahl der Netzwerkzweige, aber dennoch außerordentlich hoch. Mit einem Näherungsalgorithmus soll der Rechenaufwand im Schritt 2 drastisch reduziert werden. Dazu ordnet man dem Netzwerkgraphen $\mathcal{G}_{\mathcal{N}}$ einen schlichten Graphen G (G enthält also keine Schlingen und keine parallelen Kanten) mit gleicher Kantenmenge $\{1, \dots, b\}$ wie $\mathcal{G}_{\mathcal{N}}$ zu. Der Graph G sei vollständig, d. h. jedes Paar unterschiedlicher Knoten ist durch eine Kante verbunden. Die Kantenmenge von G sei mit E bezeichnet. Ein Zweig $e \in E$ zwischen den Knoten i und j ist wie folgt mit einem Gewicht $w(e)$ zu versehen:

1. Enthält der Netzwerkgraph $\mathcal{G}_{\mathcal{N}}$ zwischen den Knoten i und j einen Nullator- oder einen Noratorzweig, dann setze $w(e) = 1$.
2. Sind die Knoten i und j in $\mathcal{G}_{\mathcal{N}}$ durch einen Spannungsquellen- oder einen C -Zweig verbunden, dann setze $w(e) = 2$.
3. Sind die Knoten i und j in $\mathcal{G}_{\mathcal{N}}$ durch einen R -Zweig verbunden, dann setze $w(e) = 3$.
4. Sind die Knoten i und j in $\mathcal{G}_{\mathcal{N}}$ durch einen Stromquellen- oder einen L -Zweig verbunden, dann setze $w(e) = 4$.
5. Für alle noch nicht verwendeten Zweige setze ebenfalls $w(e) = 4$.

Dabei ist die jeweils erste zutreffende Regel anzuwenden. Außerdem seien $O, U \subset E$ jene Zweigmengen von G die den Norator- bzw. Nullatorzweigen von $\mathcal{G}_{\mathcal{N}}$ entsprechen. Da nur Nullator-Norator-Paare erlaubt sind, gilt $|O| = |U|$.

Eingabe:	Graph G mit den Teilmengen $O, U \subset E$.
Schritt 1:	Sortiere die Kanten in $E \setminus \{O \cup U\}$ nach aufsteigendem Gewicht: $w(e_1) \leq w(e_2) \leq \dots \leq w(e_m)$
Schritt 2:	Setze $T_1 := O, T_2 := U, i := 1$.
Schritt 3:	Falls sowohl $T_1 \cup \{e_i\}$ als auch $T_2 \cup \{e_i\}$ einen Wald bilden, dann setze $T_1 := T_1 \cup \{e_i\}$ und $T_2 := T_2 \cup \{e_i\}$.
Schritt 4:	Falls $i < m$, dann setze $i := i + 1$ und gehe zu Schritt 3.
Ausgabe:	Waldpaar (T_1, T_2) .

Tabelle 2: Näherungsalgorithmus zur Suche eines minimalen aufspannenden Baumpaares

Die Suche nach einem normalen Paar konjugierter Bäume in \mathcal{G}_N ist nun auf das Auffinden eines minimalen Baumpaares im Graphen G zurückgeführt worden, d. h. man sucht eine Kantenmenge $F \subset E \setminus \{U \cup O\}$ derart, daß sowohl $T_1 := O \cup F$ als auch $T_2 := U \cup F$ aufspannende Bäume sind und die Summe der Kantengewichte in diesen Bäumen minimiert wird:

$$\sum_{e \in T_1} w(e) = \sum_{e \in T_2} w(e) \longrightarrow \text{Min!} \quad (2)$$

Für den Spezialfall $O = U$, d. h. wenn in G ein einzelner aufspannender Baum statt eines aufspannenden Baumpaares gesucht wird bzw. wenn im Netzwerkgraphen \mathcal{G}_N Nullator-Norator-Paare nur in Parallelschaltung auftreten, kann die Optimierungsaufgabe (2) z. B. mit dem Algorithmus von KRUSKAL [11] gelöst werden. Als Verallgemeinerung dieses Verfahrens betrachte man den Algorithmus in Tabelle 2. Für den o. g. vollständigen Graphen G hat dieser Algorithmus folgende Eigenschaften:

1. Das Waldpaar (T_1, T_2) , welches nach Abarbeitung des Algorithmus vorliegt, ist ein aufspannendes Baumpaar (vgl. [9, Beweis von Lemma 1]).
2. Die Minimalität (2) kann i. allg. nicht garantiert werden. Beim Algorithmus von KRUSKAL wird die Minimalität unter Ausnutzung des *Austauschaxioms* bewiesen. Überträgt man dies auf die hier vorliegende Situation, dann müßte für beliebige Waldpaare $(O \cup F, U \cup F)$ und $(O \cup F', U \cup F')$ die Beziehung

$$|F| = |F'| + 1 \implies \exists e \in F \setminus F' : \begin{cases} O \cup F' \cup \{e\} & \text{ist ein Wald} \\ U \cup F' \cup \{e\} & \text{ist ein Wald} \end{cases}$$

gelten. Das ist aber i. allg. nicht der Fall.

3. Die Komplexität des Algorithmus beträgt $\mathcal{O}(m \log m)$. Bezogen auf die Zweiganzahl b des Netzwerkgraphen \mathcal{G}_N erhält man wegen $m = \mathcal{O}(b^2)$ einen Rechenaufwand der Ordnung $\mathcal{O}(b^2 \log b)$.

Mit dem modifizierten Algorithmus erhält man ein reguläres Index-1-Netzwerk, aber nicht notwendig mit minimaler Anzahl von Zusatzelementen. Da im Algorithmus aus Tab. 2 das aufspannende Baumpaar ausschließlich durch schrittweises Hinzufügen (also ohne Austauschen) von Kanten entstanden ist, lassen sich die Modifikationen gemäß Schritt 3 des Algorithmus aus Tab. 1 auch simultan durchführen. Bei diesem schrittweisen Vorgehen wird der Index sukzessive reduziert. Dadurch läßt sich das Verfahren ggf. auch bei einem höheren Index abbrechen, z. B. kann man sich beim Einsatz von DASSL (Differential Algebraic System Solver [2]) mit einer Reduktion auf den Index 2 zufrieden geben.

ANWENDUNGSBEISPIELE

Beispiel 1: Die in Abb. 1 dargestellte (linearisierte) Dekodierschaltung dient der Entschlüsselung chaotisch chiffrierter Signale [12]. Die Koeffizientenmatrizen des Gls. (1) sind regulär und haben

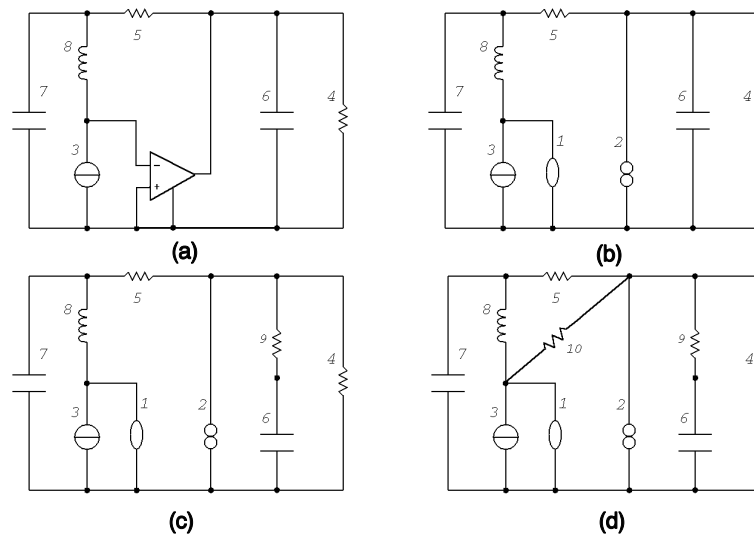


Abbildung 1: (a) Dekodierschaltung, (b) Index-4-Netzwerk, (c) Modifiziertes Index-3-Netzwerk, (d) Regularisiertes Index-1-Netzwerk

den generischen Index 4. Zu Beginn besteht t_1 nur aus dem Noratorzweig (Zweig 2) und t_2 nur aus dem Nullatorzweig (Zweig 1). Von den bestehenden Zweigen kann nur der Zweig 7 (Kapazitätszweig) zu (t_1, t_2) hinzugefügt werden. Jetzt wird der verbleibende Kapazitätszweig (Zweig 6) – der sich ohnehin nicht zu (t_1, t_2) hinzufügen läßt – um einen Widerstand oder eine Induktivität in Reihe ergänzt (entspr. Schritt 3(i) des Alg. aus Tab. 1). Das daraus resultierende Netzwerk (vgl. Abb. 1(c)) hat generisch den Index 3. Zur Konstruktion eines Paares konjugierter Bäume gibt es nur noch die in Abb. 1(d) dargestellte Möglichkeit zum Hinzufügen eines weiteren Zweiges. Das so um einen Widerstands- oder Kapazitätszweig erweiterte Netzwerk (vgl. Schritt 3(iii) des Alg. aus Tab. 1) ist generisch regulär und vom Index 1.

Beispiel 2: Betrachte sei die Komparatorschaltung aus Abb. 2(a),(b). Der Lastwiderstand R wird durch eine Brückenschaltung aus Operationsverstärkern angesteuert. Weil der Netzwerkgraph \mathcal{G}_N kein Paar konjugierter Bäume enthält, ist die Schar der Koeffizientenmatrizen $E(p)$ und $A(p)$ des Gls. (1) generisch singulär. Da \mathcal{G}_N keine Maschen aus Nullator- oder Noratorzweigen besitzt, sind die o. g. Algorithmen anwendbar. Den sukzessive Aufbau eines Paares konjugierter Bäume beginnt man mit t_1 als Menge der Noratorzweige und t_2 als Menge der Nullatorzweige. Im nächsten Schritt ist das Waldpaar (t_1, t_2) um den Spannungsquellenzweig zu ergänzen. Weitere Zweige von \mathcal{G}_N (d. h. die Widerstandszweige R, R' und R_1) können nicht hinzugefügt werden, weil sonst eine Masche entstehen würde. Abb. 2(c),(d) zeigt die zwei Möglichkeiten für zwei Zusatzzweige (R_2 und R_3), die zu einem generisch regulären Netzwerk führen.

Literatur

- [1] E. Griepentrog and R. März. *Differential-Algebraic Equations and Their Numerical Treatment*, Band 88 of *Teubner-Texte zur Mathematik*. Teubner Verlagsgesellschaft, Leipzig, 1986.
- [2] K. E. Brenan, S. L. Campbell, and L. R. Petzold. *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*. North-Holland, 1989.

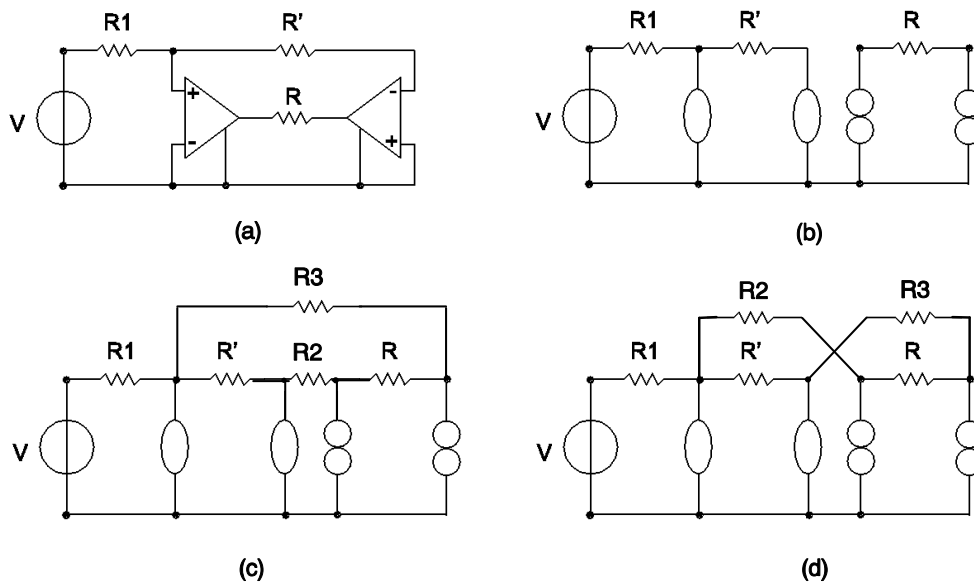


Abbildung 2: (a) Komparatorschaltung, (b) Netzwerk \mathcal{N} (c),(d) Regularisierte Netzwerke

- [3] E. Hairer, C. Lubich, and M. Roche. *The Numerical Solution of Differential-Algebraic Systems by Runge-Kutta Methods*, Band 1409 of *Lecture Notes in Mathematics*. Springer-Verlag, 1989.
- [4] G. Reißig. An extension of the normal tree method. In *Proc. Europ. Conf. Circ. Th. Design (ECCTD), Budapest, Hu., Aug. 30–Sept. 3, 1997*.
- [5] K. Weierstrass. Zur Theorie der bilinearen und quadratischen Formen. In *Monatsbericht der Preussischen Akademie der Wissenschaften*, Berlin, 1868. Reprinted in: *Mathematische Werke von Karl Weierstrass*, Band II, Mayer & Müller, Berlin, 1895, S. 310-338.
- [6] F. R. Gantmacher. *Matrizentheorie*. Springer Verlag, 1986.
- [7] M. Hasler. Non-linear non-reciprocal resistive circuits with a structurally unique solution. *Int. J. Cir. Theo. Appl.*, 14:237–262, 1986.
- [8] M. Fosséprez. *Non-linear Circuits - Qualitative Analysis of Non-linear, Non-reciprocal Circuits*. Wiley, 1992.
- [9] K. Röbenack and G. Reißig. Index one regularization of networks using a minimum number of additional elements. In *Proc. SDA'98 (Workshop on System Design Automation)*, S. 144–151, 1998.
- [10] G. Reißig and K. Röbenack. Eine neue Methode zum Auffinden von Paaren konjugierter Bäume. In *Kleinheubacher Berichte 1996, Band 40*, S. 559–565. Deutsche Telekom, 1997.
- [11] J. B. Kruskal. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proc. Amer. Math. Soc.*, 7:48–50, 1956.
- [12] U. Feldmann, M. Hasler, and W. Schwarz. On the design of a synchronizing inverse of a chaotic system. In *Proc. Europ. Conf. Circ. Th. Design (ECCTD), Istanbul, Turkey, Band 1*, S. 479–482, 1995.

Autorenangaben:

Dipl.-Ing. Röbenack, K.

Dipl.-Ing. Dipl.-Math. Reißig, G.

Institut für Regelungs- und Steuerungstheorie (Prof. Dr. K. Reinschke)

Technische Universität Dresden, Mommsenstr. 13

01062 Dresden

Tel.: +49 351 3940

Fax.: +49 351 7281, E-mail: roebenak@erss11.et.tu-dresden.de, reiszig@erss11.et.tu-dresden.de