

# Ketten evolutionärer Prozessoren mit Random-Context-Filtern\*

Bianca Truthe

Otto-von-Guericke-Universität Magdeburg, Fakultät für Informatik  
PSF 4120, D-39016 Magdeburg  
truthe@iws.cs.uni-magdeburg.de

**Zusammenfassung** Es wird gezeigt, dass jede rekursiv aufzählbare Sprache von einem akzeptierenden Netz evolutionärer Prozessoren (ANEP) in der Form einer Kette (jeder Prozessor hat höchstens zwei Nachbarn), eines Ringes (jeder Prozessor hat genau zwei Nachbarn) oder eines Rades (ein Ring mit einem zusätzlichen Mittelknoten) akzeptiert wird, wobei die Kommunikation zwischen den Prozessoren durch Filter gesteuert wird, die die Ab- oder Anwesenheit gewisser Buchstaben in den zu kommunizierenden Wörtern überprüfen. Die Größe der konstruierten Netze hängt nicht von der akzeptierten Sprache ab. Hiermit werden einige Fragen beantwortet, die von J. Dassow und F. Manea in der auf der Konferenz „Descriptive Complexity of Formal Systems“ (DCFS) 2010 vorgestellten Arbeit ([3]) offen gelassen wurden.

## 1 Einleitung

Netze von Sprachprozessoren wurden von E. CSUHAJ-VARJÚ und A. SALOMAA eingeführt ([2]). Solch ein Netz kann als Graph angesehen werden, bei dem jeder Knoten Regeln und Wörter hat, die er entsprechend den Regeln ableitet, und die nach dem Passieren gewisser Filter über die Kanten zu anderen Knoten gelangen.

Von Punktmutationen in der Biologie inspiriert, haben J. CASTELLANOS, C. MARTÍN-VIDE, V. MITRANA und J. SEMPERE in [1] Netze evolutionärer Prozessoren eingeführt. Dabei sind die verwendeten Regeln Ersetzen eines Buchstabens durch einen anderen, Einfügen eines Buchstabens und Löschen eines Buchstabens.

Akzeptierende Netze evolutionärer Prozessoren wurden von M. MARGENTERN, V. MITRANA und M. J. PÉREZ-JIMÉNEZ in [4] eingeführt. In einem solchen Netz gibt es einen sogenannten Eingabe-Knoten, der zu Beginn der Berechnung das zu untersuchende Wort (Eingabewort) enthält (alle anderen Knoten enthalten keine Wörter), und einen oder mehrere sogenannte Ausgabe-Knoten. Ein Eingabewort wird genau dann akzeptiert, wenn ein (oder jeder) Ausgabe-Knoten irgendwann im Laufe der Berechnung ein Wort enthält. Die Kommunikation zwischen den Prozessoren wird durch Filter gesteuert, die die Ab- oder

---

\* Die Arbeit wurde unter dem Titel *Chains of Evolutionary Processors with Random Context Filters* auf der Konferenz NCMA 2013 vorgestellt [5].

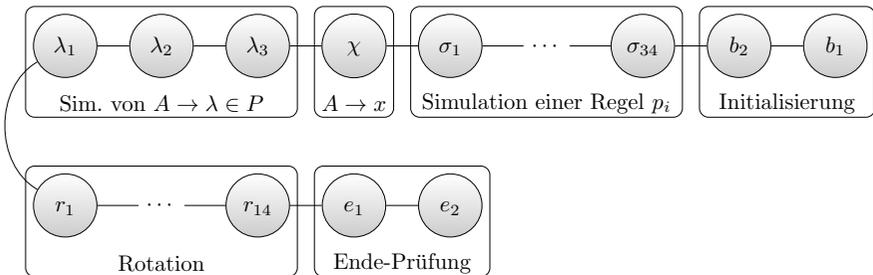
Anwesenheit gewisser Buchstaben in den zu kommunizierenden Wörtern überprüfen (Random-Context-Filter). Dabei wird zwischen Eingangs- und Ausgangsfiltern unterschieden. Jene Wörter die sich in einem Knoten  $K$  befinden, werden zu Beginn eines Kommunikationsschrittes vom Ausgangsfilter des Knotens  $K$  überprüft. Erfüllt ein Wort die Filterbedingungen, verlässt es den Knoten  $K$  und wird zu allen Nachbarn gesendet, andernfalls bleibt es im Knoten  $K$  und wird dort weiter verarbeitet. Jene Wörter, die von einem Knoten  $K$  zu einem Knoten  $K'$  übermittelt werden, werden vom Eingangsfilter des Knotens  $K'$  überprüft. Erfüllt ein Wort die Filterbedingungen, gelangt es in den Knoten  $K'$ , andernfalls geht es verloren.

In der Arbeit [3] haben J. DASSOW und F. MANEA Untersuchungen zu akzeptierenden Netzen mit speziellen Formen hinsichtlich ihrer Mächtigkeit und Komplexität veröffentlicht. Insbesondere wurden Netze betrachtet, die die Form eines Sternes, Rades oder Gitters haben. In einem Stern gibt es einen zentralen Knoten, der mit jedem anderen Knoten in beide Richtungen verbunden ist, aber keine weiteren Kanten. In einem Rad gibt es einen zentralen Knoten, der mit jedem anderen Knoten in beide Richtungen verbunden ist, und jeder andere Knoten hat neben dem zentralen Knoten genau zwei weitere Nachbarn; der zentrale Knoten ist der Ausgabe-Knoten. Zu einem Gitternetz gibt es zwei natürliche Zahlen  $m$  und  $n$  und eine umkehrbar eindeutige Abbildung der Knoten auf die Paare  $(i, j)$  für  $1 \leq i \leq m$  und  $1 \leq j \leq n$  so, dass die Kanten gerade die Verbindungen zwischen den Knoten mit den Markierungen  $(i, j)$  und  $(i+1, j)$ ,  $(i-1, j)$ ,  $(i, j+1)$  und  $(i, j-1)$  darstellen (sofern  $1 \leq i \pm 1 \leq m$  und  $1 \leq j \pm 1 \leq n$  gilt).

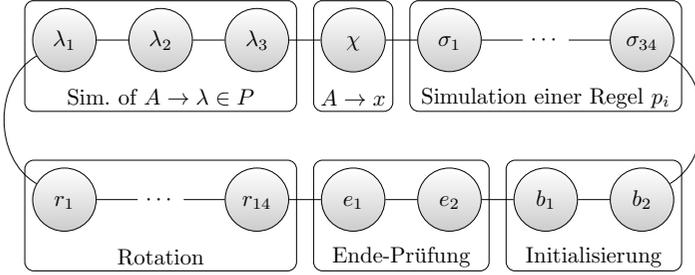
In der vorliegenden Arbeit werden die Untersuchungen fortgesetzt und einige offene Fragen aus der Arbeit [3] geklärt.

## 2 Ergebnisse

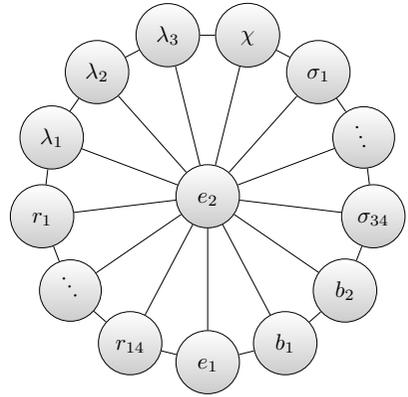
Jede rekursiv aufzählbare Sprache wird durch eine Kette von 56 evolutionären Prozessoren mit Random-Context-Filtern akzeptiert. Zu einer Grammatik in Kuroda-Normalform kann ein Netz mit folgender Struktur konstruiert werden, das genau die Sprache akzeptiert, die von der simulierten Grammatik erzeugt wird (die Linien stehen für Kanten in beide Richtungen):



Auf Grund des Eingangsfilters vom Ausgabe-Knoten  $N_{e_2}$  ändert ein Verbinden der Knoten  $N_{e_2}$  und  $N_{b_1}$  nichts an der akzeptierten Sprache, so dass die gleiche Sprache auch von einem Ring mit 56 Knoten akzeptiert wird:



Auch das Umformen zu dem nebenstehenden Rad ändert nichts an der akzeptierten Sprache. Der Ausgabe-Knoten  $N_{e_2}$  erwartet ein Symbol, das nur im Knoten  $N_{e_1}$  erzeugt wird. Folglich gelangen keine Wörter über die anderen Kanten hin zum Ausgabe-Knoten. Auch zwischen dem Eingabe-Knoten  $N_{b_1}$  und dem Knoten  $N_{e_1}$  findet keine direkte Kommunikation statt, da der Knoten  $N_{b_1}$  nur am Anfang ein Wort enthält und in diesem kein Hilfssymbol vorkommt, was aber vom Knoten  $N_{e_1}$  erwartet wird, und jedes Wort im Knoten  $N_{e_1}$  ein Hilfssymbol enthält, womit es nicht in den Knoten  $N_{b_1}$  gelangt.



## Literatur

1. J. Castellanos, C. Martín-Vide, V. Mitrana, and J. M. Sempere. Solving NP-Complete Problems With Networks of Evolutionary Processors. In *IWANN 2001*, volume 2084 of *LNCS*, pages 621–628. Springer, 2001.
2. E. Csuhaj-Varjú and A. Salomaa. Networks of Parallel Language Processors. In *New Trends in Formal Languages – Control, Cooperation, and Combinatorics*, volume 1218 of *LNCS*, pages 299–318. Springer, 1997.
3. J. Dassow and F. Manea. Accepting Hybrid Networks of Evolutionary Processors with Special Topologies and Small Communication. In *DCFS 2010*, volume 31 of *EPTCS*, pages 68–77, 2010.
4. M. Margenstern, V. Mitrana, and M. J. Pérez-Jiménez. Accepting Hybrid Networks of Evolutionary Processors. In *10th Intern. Workshop on DNA Computing*, volume 3384 of *LNCS*, pages 235–246. Springer, 2004.
5. B. Truthe. Chains of Evolutionary Processors with Random Context Filters. In *NCMA 2013*, books@ocg.at. Österreichische Comp. Ges., Austria, 2013.