

Proseminar (3 LP)

Zellularautomaten und diskrete komplexe Systeme
im Sommersemester 2019

Ausarbeitung

von **Klaus Philipp Theyssen**, Matr.nr. 2061578

Thema

Ferdinand Peper and Jia Lee (2018)
On Non-polar Token-Pass Brownian Circuits
Reversibility and Universality, S.299-311

Erklärung

gemäß §6 (11) der Prüfungsordnung Informatik (Bachelor) 2015

Ich versichere wahrheitsgemäß, die Seminararbeit zum Proseminar „Zellularautomaten und diskrete komplexe Systeme“ im Sommersemester 2019 selbstständig angefertigt, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Abänderungen entnommen wurde.

(Klaus Philipp Theyssen, Matr.nr. 2061578)

1 Einführung

Bei Elektronik im Nanometer-Bereich sind Rauschen und Fluktuation entscheidende Faktoren die beim Entwurf entsprechender Schaltkreise zu beachten sind. Desweiteren ist ein geringer Energieverbrauch anzustreben, daher könnten in Zukunft Schaltkreise von Interesse sein die nur von einzelnen Partikeln geschaltet werden.

Die in dem Aufsatz [4] präsentierten brownischen Schaltkreise nutzen Tokens als Signale und setzen Fluktuation aktiv bei ihren Berechnungen ein. Die Fluktuation ist angelehnt an die brownische Molekularbewegung in der Biologie.

In [1] werden brownische Schaltkreise auf Basis von Petri-Netzen eingeführt und entsprechend formalisiert. Die in [1] vorgestellten Tokens der brownischen Schaltkreise haben allerdings implizit einen Bias in eine Richtung.

In [4] werden nichtpolare token-pass brownische Schaltkreise vorgestellt, diese haben den Bias nicht.

Die grundlegende Idee ist, dass durch den zusätzlichen Nichtdeterminismus weniger Funktionalität explizit modelliert werden muss und somit weniger Bauteile verwendet werden können.

2 Grundlagen

Zunächst werden die im Aufsatz behandelten Schaltkreis Typen vorgestellt um Unterschiede in Funktionalität und Aufbau hervorzuheben. Dann wird das T-Element betrachtet und wie sich damit die Universalität der brownischen token-pass Schaltkreise ergibt.

Tokens sind diskrete nicht teilbare Einheiten die Signale modellieren. Alle hier vorgestellten Schaltkreise sind asynchron, dies bedeutet sie haben keinen Zeitgeber und es kann nebenläufig zu Änderungen am Signal kommen. Sie sind robust gegen Verzögerungen (delay-insentive), was heißt, dass Verzögerungen in der Signalweiterleitung nicht zu inkorrekten Berechnungen führen.

2.1 Token-based Schaltkreise

In token-basierten Schaltkreise werden Signale als einzelne diskrete Token (Partikel) auf den Kabeln modelliert. Es gibt keine weiteren Einschränkungen bezüglich Token und Tokenweiterleitung, z.B. dürfen Tokens erzeugt werden und die Kabel wechseln.

Ein Beispiel für token basierte Schaltkreise sind Petri-Netze.

In [1] wird die formale Definition von Petri-Netze auf brownische Schaltkreise übertragen.

Token basierte Schaltkreise die delay insentive sind können mit einer Menge von Schaltkreisprimitiven konstruiert werden, genauso wie synchrone Schaltkreise aus NOT-Gattern und UND-Gattern [4]. Eine solche Menge nennt man dann universell und Abbildung 1 gibt dafür ein Beispiel.

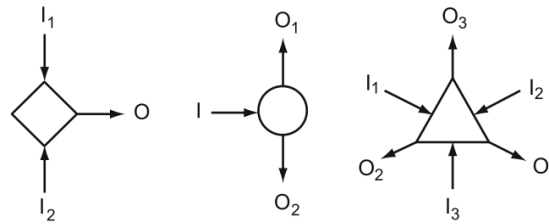


Abbildung 1: Merge, Fork und Tria

Dabei führt Merge zwei Kabel zu einem zusammen, wobei die Tokens einfach nur weitergeleitet werden. Fork macht aus einem Token zwei. Tria fügt zwei Tokens zusammen und je nach Eingabekabel kommt die Ausgabe auf ein bestimmtes Kabel. Für jedes Input Token I_i ($i \in \{1, 2, 3\}$) und I_j ($j \in \{1, 2, 3\}$) mit $i \neq j$ erhalten wir das Ausgabe Token auf O_{6-i-j} . Wobei das Zusammenführen beim Tria nur funktioniert wenn zwei Tokens da sind, ein einzelnes Token wartet solange bis ein zweites kommt. Diese Funktionalität in asynchronen Schaltkreisen hat die Aufgabe die verschiedenen nebenläufigen Berechnungen zu synchronisieren und entspricht gewissermaßen dem Takt in synchronen Schaltkreises und ist deshalb hervorzuheben.

2.2 Token-pass Schaltkreise

Der Name kommt von der Bauweise dieser Schaltkreise, sie verbinden einfach nur Kabel miteinander durch die Tokens hindurchlaufen und die Schaltkreiselemente geben diese weiter oder nicht.

Token-pass Schaltkreisen lassen die Zahl der Tokens gleich. Tokens können nicht entstehen oder verschwinden und auch nicht auf andere Kabel wechseln.

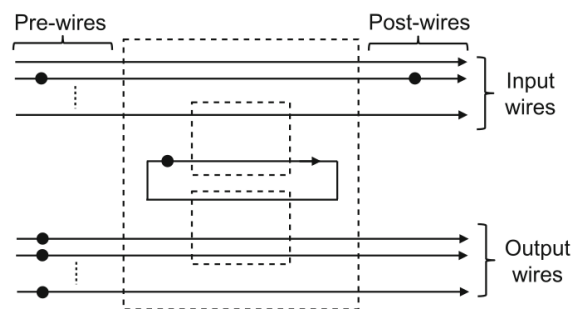


Abbildung 2: Token-pass Schema

Token-pass Schaltkreise haben eine Menge an Eingabekabeln die in den Schaltkreis führen und eine Menge an Ausgabekabeln. Wobei diese in einen Abschnitt vor dem Schaltkreis (pre-wire) und einen Abschnitt danach (post-wire) eingeteilt werden. Die Eingabekabel können im pre-wire Abschnitt beliebig Token haben

die aber wenn vorhanden auch auf den post-wire Abschnitt geführt werden. Die Ausgabekabel haben im pre-wire Abschnitt alle ein Token und haben dann beliebig Tokens auf dem post-wire Abschnitt, entsprechend der Ausgabe. Innerhalb des Schaltkreises können sich Schleifen befinden.

2.3 Von token-basiert nach token-pass

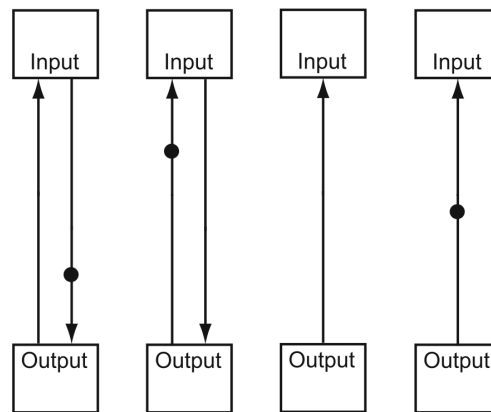


Abbildung 3: token-pass (links) und token-basiert (rechts)

Token-basierte Schaltkreise können äquivalent als token-pass Schaltkreise dargestellt werden, indem einfach aus jedem Kabel zwei werden und die Token sich entsprechend der Abbildung 3 verhalten.

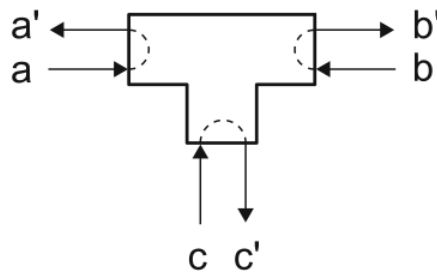
2.4 Brownsche Schaltkreise

Brownsche Schaltkreise sind eine Erweiterung der token-pass Schaltkreise. Tokens können sich nun frei bewegen, angelehnt an die brownsche Molekularbewegung. Diese Fluktuation ist treibende Kraft für das Zusammenwirken der Tokens innerhalb der Schaltkreiselemente. Sie ermöglicht Tokens aus Sackgassen wieder zu entkommen, was sich in einfacherem Design widerspiegelt da nicht alles explizit modelliert werden muss.

2.4.1 Polare token-pass brownsche Schaltkreise

In polaren token-pass Schaltkreises existiert eine bevorzugte Richtung (Bias) der Token, gekennzeichnet durch einen Pfeil.

Polares T-Element Grundlegende Funktion des T-Elementes entspricht. Eingang c ist der Basis Eingang des T-Elementes, hier muss immer ein Token anliegen damit es zur Verarbeitung kommt. Wenn bei c ein Token anliegt und bei einem der beiden anderen Eingänge a oder b auch ein Token anliegt werden diese vom



Die nicht-polaren Schaltkreise ermöglichen einfacheres Design und Verwendung von weniger T-Elementen, weil bestimmtes Verhalten zum Verhindern von Deadlocks, ausgelöst von Sackgassen, nicht explizit modelliert werden muss.

3 Universalität des T-Element

In Abbildung 6 ist zu erkennen wie die token-basierten Schaltkreispimitive (Merge, Fork und Tria) mit mithilfe des T-Elementes nachgebaut werden.

Daraus folgt, dass das brownsche T-Element universell für die Klasse der token-pass brownschen Schaltkreise ist.

Laut den Autoren ist dies sogar die Klasse aller berechenbarer Funktionen und somit sind token-pass brownsche Schaltkreise berechnungsuniversell. Die Berechnungsuniversalität der token-basierten Schaltkreise wird in [2] gezeigt, dies geschieht allerdings durch Verweisen auf noch ältere Aufsätze bis schließlich auf [3] verwiesen wird. Dadurch kann ich nicht ganz nachvollziehen ob man wirklich von berechnungsuniversell sprechen kann bzw. die Universalität von einer Implentierung erreicht werden würde.

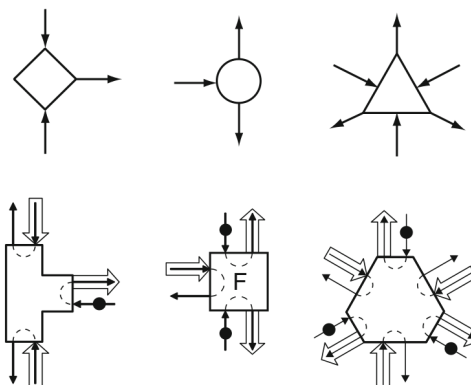


Abbildung 6: Äquivalenz token-based und token-pass

Das direkte Nachbauen von token-basierten Schaltkreisen mithilfe der TP-Merge, TP-Fork und TP-Tria ist nicht effizient. Da dabei das Fluktuieren der Tokens nicht ausgenutzt wird. Beispielsweise benötigt ein 1-Bit Speicher der naiv mithilfe der Schaltkreisp primitiven nachgebaut ist insgesamt 26 T-Elemente. Im nächsten Abschnitt werden wir sehen, dass dies sehr viel effizienter möglich ist, wenn man Eigenschaften der brownschen token-pass Schaltkreise beim Design richtig ausnutzt.

4 1-Bit Speicher

Nun soll anhand eines 1-Bit Speichers die Funktionsweise von brownschen token-pass Schaltkreisen erläutert werden. Mithilfe von polaren T-Elementen ist es

möglich einen 1-Bit Speicher mit 8 T-Elementen zu bauen [1]. Bei nicht-polaren brownischen T-Elementen sind es sogar nur 7 T-Elemente [4].

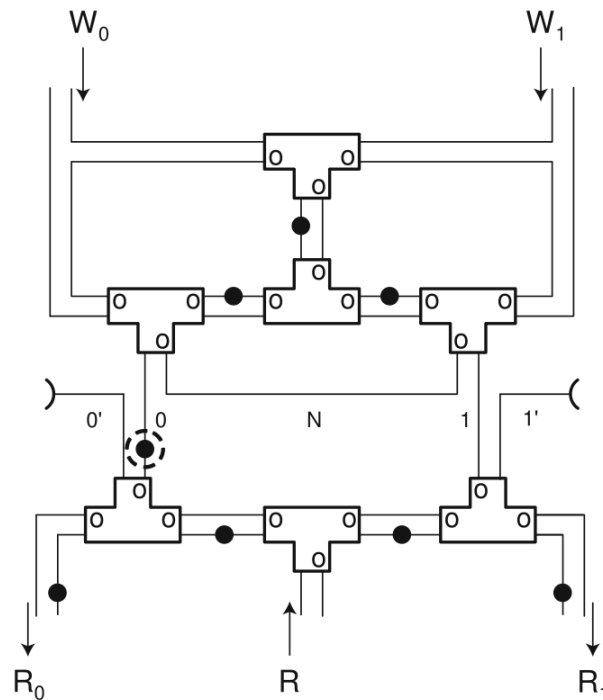


Abbildung 7: 1-Bit Speicher nicht polar token pass

Grundlegend gibt es jeweils eine Menge an T-Elementen für das Lesen bzw. Schreiben. Von zentraler Bedeutung ist ein Token in der Mitte des Schaltkreises, das über seine Position den derzeitigen Zustand des Speichers angibt. Im folgenden wird dieses Token mit Zustandstoken bezeichnet.

Es kann die folgende Zustände einnehmen:

- 0 eine 0 ist gespeichert
- 1 eine 1 ist gespeichert
- N neutraler Zustand, der eingenommen wird bevor neuer Zustand geschrieben wird
- 0' temporärer Zustand während 0 ausgelesen wird
- 1' temporärer Zustand während 1 ausgelesen wird

Der Speicher soll mit einer 0 initialisiert sein. Entsprechend ist auch das Zustandstoken (umkreist) in Abbildung 7 positioniert.

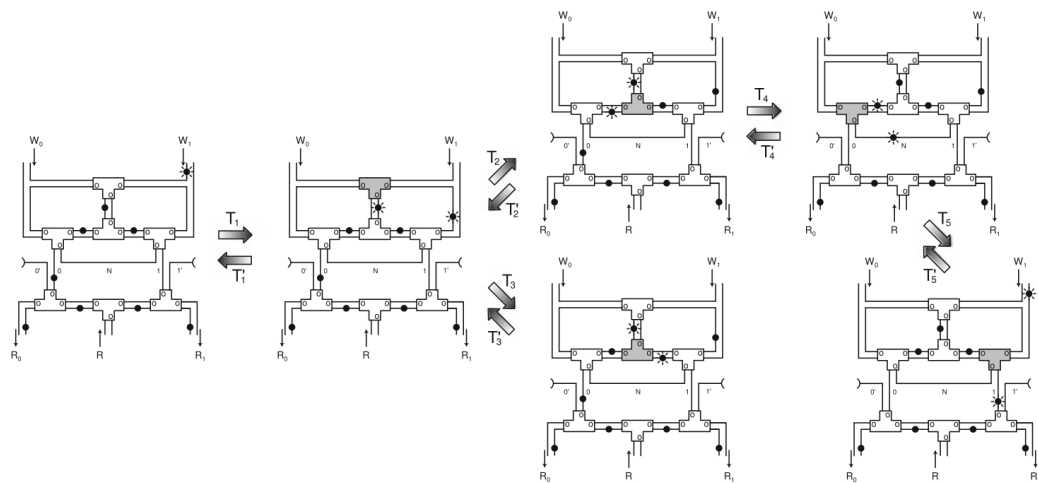


Abbildung 8: Schreiben einer 1

4.1 Schreibvorgang

Zunächst soll der Schreibvorgang erläutert werden. Wobei eine 0 gespeichert ist und eine 1 geschrieben wird. Der Schaltkreis durchläuft 6 verschiedenen Zustände, wobei einer davon eine Sackgasse ist. In Abbildung 8 sind die entsprechenden Übergänge gekennzeichnet und jeweils das T-Element, was als letztes Tokens bewegt hat, ist grau markiert. Als Eingabe haben wir ein Token auf W_1 , dieses wird entsprechend dem Übergang T_1 in Abbildung 8 verarbeitet. Jetzt kann das Token unterhalb des graumarkierten T-Elementes entweder nach rechts T_3 oder links T_2 wandern. Wobei rechts in eine Sackgasse führt, wenn eine 0 gespeichert ist. Nach T_2 kann im nächsten Schritt T_3 ausgeführt werden womit das Zustandstoken in den Zustand N überführt wird. Anschließend kann es über T_5 mit dem ursprünglichen Eingabetoken interagieren und so den neuen Zustand 1 annehmen.

Zu jedem Übergang $T_i, i \in \{1, 2, 3, 4, 5\}$ gibt es einen entsprechenden Übergang T'_i der ihn wieder rückgängig macht.

4.2 Lesevorgang

Jetzt wird das Auslesen einer gespeicherten 1 erläutert. Als Eingabe haben wir ein Token auf R. Dieses kann nun dem mit dem Token links T_6 oder rechts T_7 vom ersten T-Element verarbeitet werden. Wenn eine eins gespeichert ist führt links in eine Sackgasse, aus der die Berechnung mit T'_6 wieder entkommt. Nach T_7 kann das Zustandstoken über T_8 in den Zustand 1' überführt werden. Hier kann das Zustandstoken dann via T_9 mit dem Ausgabetoken interagieren. Wie beim Schreiben kann jeder Berechnungsschritt durch einen entsprechenden Übergang wieder rückgängig gemacht werden. Da jedoch das Ausgabekabel polar ist (gekennzeichnet durch Pfeil) ist die Wahrscheinlichkeit das nach Interaktion mit dem

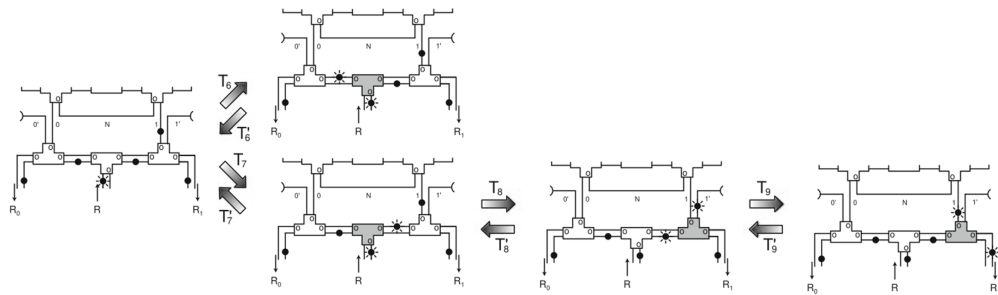


Abbildung 9: Lesen einer 1

Ausgabetoken die Berechnung rückgängig gemacht wird sehr gering.

5 UND-Bauteil

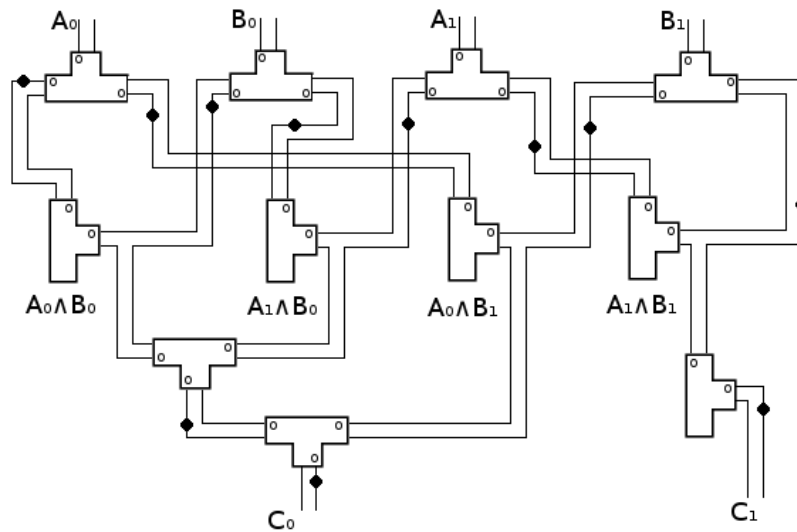


Abbildung 10: UND-Gatter aus 11 T-Elementen

Als Teil meiner Eigenarbeit im Rahmen dieses Proseminars habe ich ein UND-Gatter mithilfe von nicht-polaren T-Elementen entworfen.

Es benutzt den Suchmechanismus der Tokens und Backtracking falls sie in eine Sackgasse laufen. Die Idee ist jede mögliche Eingabekombination jeweils mit einem logischen UND zu verknüpfen. Das erste T-Element jeder Eingabe wird zum Modellieren der möglichen zwei Wege benutzt. Die beiden Eingabe Token versuchen sich in dem der Eingabe entsprechenden T-Element, welches dem UND entspricht, zu finden. Haben sich beide Input Tokens gefunden wird ein Token

weitergeleitet. Bei der Eingabe (A_1, B_1) wird dieses Token direkt zu Ausgabe C_1 , bei allen anderen Eingabemöglichkeiten werden die Token mithilfe von zwei weiteren T-Elementen zur Ausgabe C_0 zusammengeführt.

5.1 Initialisierung

Es ist eine Initialisierung auf Abbildung 10 gegeben (die Position der Tokens), solange mit dieser gestartet wird und nur gültige Eingaben (2 Tokens) vorkommen, ist die Berechnung korrekt. Diese Initialisierung ist nicht eindeutig und auch die Anordnung der Elemente ist veränderbar, was im Hinblick auf möglichst kurze Kabel für eine schnellere Berechnung von Interesse ist bzw. zu optimieren wäre. Sobald jedoch die Initialisierung zu Beginn nicht gegeben ist kann es zu falschen Ausgaben kommen wie: 0 und 1 gleichzeitig oder zweimal 0 bzw. 1.

6 Ausblick

6.1 Geschwindigkeit der Berechnung

Die Fluktuation der Tokens auf einem Kabel der Länge L führt zu erwarteten Zeit von $O(L^2)$ in der ein Token diese Strecke zurücklegt. Folglich ist die Länge der Kabel möglichst minimal zu halten. Damit ist die Frage nach der optimalen Anordnung der Bauteile und der damit verbundenen Initialisierung auch von Bedeutung. Desweiteren kann man Sperren einsetzen, sodass Tokens auf bestimmten Kabeln sich nur in eine Richtung bewegen können. Konkret kann dies in Form von Dioden realisiert werden, diese verbrauchen allerdings Strom [1].

6.2 Implementierung

Das T-Element ist für theoretische Überlegungen geeignet jedoch für eine Implementierung nicht optimal. Es hat zu viele Kabel und ist zu komplex um sinnvoll als Schaltkreissprimitiv eingesetzt zu werden. Es gibt noch einfachere Schaltkreiselemente [1] allerdings funktionieren diese nur mit Fluktuation. Was im Hinblick auf eine Implementierung im Prinzip nicht umzusetzen ist [1]. Auch kann es zu Interaktionen zwischen Tokens kommen die bei diesem Modell nicht beachtet werden. (z.B. Elektronen als Tokens und entsprechende Elektromagnetische Felder) Daher sind Forderungen wie z.B., dass Token nur auf einem Kabel bleiben dürfen bei einer konkreten Implementierung nicht leicht umzusetzen.

7 Zusammenfassung

In dem Aufsatz [4] wird eine neue Art von Schaltkreis vorgestellt der zukünftig in der Nanoelektronik eingesetzt werden könnte. Aufbauend auf polaren brownischen token-pass Schaltkreisen aus [1] werden nichtpolare Kabel und T-Elemente eingeführt, deren Token keinen Bias in eine Richtung haben. Dabei ist das Konzept

von brownischer Bewegung der Tokens und der damit verbundene Nichtdeterminismus interessant und resultiert in einer neuen Art von Schaltkreisen, deren Design und Eigenschaften weiter zu untersuchen sind. Beim Schaltkreisentwurf sind die Geschwindigkeit der Berechnung, Energieverbrauch und die Suchfähigkeit der Token alles wichtige Metriken, die man gegeneinander abwegen muss um ein gutes Resultat zu erhalten. Jedoch ist die Frage nach Arten von konkreten Implementierungen noch weitgehend offen.

Literatur

- [1] Peper F., Lee J., Carmona J., Cortadella J., and Morita K. Brownian circuits: Fundamentals. *ACM J. Emerg. Technol. Comput. Syst.*, 9(1):1–24, 2013.
- [2] Lee J., Peper F., Adachi S., and Mashiko S. Universal delay-insensitive systems with buffering lines. *IEEE Trans. Circuits Syst. I*, 52(4):742–754, 2005.
- [3] Robert M. Keller. Towards a theory of universal speed independent modules. *IEEE Transactions on Computers*, 23(1):21–32, 1974.
- [4] Ferdinand Peper and Jia Lee. On non-polar token-pass brownian circuits. In Ivan Zelinka, Andrew Adamatzky, and Guanrong Chen, editors, *Reversibility and Universality*, volume 30 of *ECC*, pages 299–312. Springer, 2018.