Reconfigurable Broadcast Networks and Asynchronous Shared-Memory Systems are Equivalent

A. R. Balasubramanian, Chana Weil-Kennedy

Technical University of Munich
Two Models

Reconfigurable Broadcast Network (RBN)

• introduced in [Delzanno, Sangnier & Zavattaro, CONCUR ’10]
• anonymous, identical processes which can communicate by selective broadcast.

Asynchronous Shared Memory System (ASMS)

• introduced in [Esparza, Ganty & Majumdar, CAV ’13]
• anonymous, identical processes which can communicate by writing to a shared register.
RBN

init

sent

a_1

a_2

a_3

b_1

b_2

b_3

final
RBN and ASMS are Equivalent, C. Weil-Kennedy
Goal: put a process in final

RBN and ASMS are Equivalent, C. Weil-Kennedy
Goal: put a process in $\text{final}$
RBN and ASMS are Equivalent, C. Weil-Kennedy

**Goal:** put a process in \textit{final}
RBN and ASMS are Equivalent, C. Weil-Kennedy

Goal: put a process in final
RBN

Goal: put a process in \textit{final}

\begin{itemize}
\item \textbf{init}
\item \textbf{sent}
\end{itemize}
RBN

Goal: put a process in $\text{final}$
RBN and ASMS are Equivalent, C. Weil-Kennedy

Goal: put a process in \textit{final}
RBN and ASMS are Equivalent, C. Weil-Kennedy

Goal: put a process in \textit{final}

\begin{itemize}
  \item \textit{init}
  \item \textit{sent}
  \item \textit{final}
\end{itemize}
RBN and ASMS are Equivalent, C. Weil-Kennedy

Goal: put a process in \textit{final}
Goal: put a process in \textit{final}
RBN

Goal: put a process in final

RBN and ASMS are Equivalent, C. Weil-Kennedy
RBN and ASMS are Equivalent, C. Weil-Kennedy

Goal: put a process in \textit{final}
ASMS

RBN and ASMS are Equivalent, C. Weil-Kennedy
RBN and ASMS are Equivalent, C. Weil-Kennedy
Goal: put a process in *final*

RBN and ASMS are Equivalent, C. Weil-Kennedy
RBN and ASMS are Equivalent, C. Weil-Kennedy

Goal: put a process in \textit{final}
RBN and ASMS are Equivalent, C. Weil-Kennedy

**Goal:** put a process in \textit{final}
ASMS

Goal: put a process in final

\[ W(1) \xrightarrow{W(2)} W(2) \xrightarrow{R(2)} R(1) \xrightarrow{q_2} \text{final} \]
Goal: put a process in final

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS

Goal: put a process in final

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS

Goal: put a process in $\text{final}$

$\begin{align*}
\text{q}_1 & \xrightarrow{W(1)} & \text{q}_2 \\
\text{q}_2 & \xrightarrow{R(1)} & \text{q}_2 \\
\text{q}_2 & \xrightarrow{W(2)} & \text{q}_2 \\
\text{q}_2 & \xrightarrow{R(2)} & \text{final}
\end{align*}$
ASMS

Goal: put a process in \textit{final}

\[ W(1) \quad W(2) \quad R(1) \quad R(2) \]

1

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS

Goal: put a process in $\text{final}$

$W(1)$

$q_1$

$W(2)$

$q_2$

$R(1)$

$R(2)$

$\text{final}$

$2$
Goal: put a process in $final$

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS

Goal: put a process in final

\[
W(1) \quad W(2) \quad R(1) \quad R(2)
\]

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS simulates RBN

Simulation of polynomial-size with bijection between configurations
ASMS simulates RBN

Simulation of polynomial-size with bijection between configurations

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS simulates RBN

Simulation idea #1

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS simulates RBN

**Simulation idea #1**

```
init

W(1)  -->  a_1

sent

W(1)  -->  a_2

W(2)  -->  R(1)  -->  a_3

W(2)  -->  R(1)  -->  a_1

W(2)  -->  R(2)  -->  a_2

W(2)  -->  R(2)  -->  a_3

W(2)  -->  R(2)  -->  b_2

W(3)  -->  R(2)  -->  b_3

b_3  -->  final

*```

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS simulates RBN

Simulation idea #1

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS simulates RBN

Simulation idea #1

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS simulates RBN

Simulation idea #1

init

sent

W(1)

W(2)

R(1)

R(1)

R(2)

R(2)

W(3)

final

1

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS simulates RBN

Simulation idea #1

init

W(1)

sent

W(1)

W(2)

R(1)

R(1)

R(1)

R(2)

R(2)

R(2)

W(3)

final

1
ASMS simulates RBN

Simulation idea #1

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS simulates RBN

Simulation idea #1
ASMS simulates RBN

Simulation idea #1

\[ \begin{array}{c}
\text{init} \\
W(1) \\
\text{sent} \\
W(1)
\end{array} \quad \begin{array}{c}
a_1 \rightarrow R(1) \quad b_1 \rightarrow R(2) \\
W(2) \\
a_2 \rightarrow R(1) \\
W(2) \\
a_3 \rightarrow R(2) \\
W(2) \\
b_2 \rightarrow R(2) \\
W(2) \\
b_3 \rightarrow W(3)
\end{array} \quad \begin{array}{c}
\text{final}
\end{array} \]
ASMS simulates RBN

Simulation idea #1

\[ R(1) \Rightarrow a_1 \]
\[ W(2) \]
\[ R(1) \Rightarrow a_2 \]
\[ R(2) \Rightarrow b_1 \]
\[ R(2) \Rightarrow b_2 \]
\[ R(2) \Rightarrow b_3 \]
\[ W(3) \]

init

sent

final

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS simulates RBN

Simulation idea #1

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS simulates RBN
ASMS simulates RBN

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS simulates RBN

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS simulates RBN

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS simulates RBN

\[ \begin{align*}
& \text{init} \\
& \text{sent} \\
& \text{stuck…} \\
& \text{final} \\
\end{align*} \]
ASMS simulates RBN

Simulation idea #1

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS simulates RBN

Simulation idea #2

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS simulates RBN

Simulation idea #2

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS simulates RBN

Simulation idea #2

$W(1) \rightarrow a_1 \rightarrow R(1) \rightarrow W(\#) \rightarrow a_2 \rightarrow R(1) \rightarrow W(\#) \rightarrow a_3 \rightarrow W(2) \rightarrow W(\#) \rightarrow b_2 \rightarrow R(2) \rightarrow W(\#) \rightarrow b_3 \rightarrow W(3) \rightarrow W(\#) \rightarrow final$

$W(1) \rightarrow W(\#) \rightarrow init \rightarrow W(\#) \rightarrow sent$
ASMS simulates RBN

Simulation idea #2

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS simulates RBN

Simulation idea #2

\[
\begin{align*}
W(1) & 
\rightarrow & 
W(\#) & 
\rightarrow & 
\text{init} \\
W(#) & 
\rightarrow & 
W(1) & 
\rightarrow & 
\text{sent} \\
R(1) & 
\rightarrow & 
W(\#) & 
\rightarrow & 
a_1 \\
R(2) & 
\rightarrow & 
W(\#) & 
\rightarrow & 
a_2 \\
W(3) & 
\rightarrow & 
W(\#) & 
\rightarrow & 
a_3 \\
R(2) & 
\rightarrow & 
W(\#) & 
\rightarrow & 
b_1 \\
W(\#) & 
\rightarrow & 
W(\#) & 
\rightarrow & 
b_2 \\
W(\#) & 
\rightarrow & 
W(\#) & 
\rightarrow & 
b_3 \\
\end{align*}
\]
ASMS simulates RBN

Simulation idea #2

RBN and ASMS are Equivalent, C. Weil-Kennedy
ASMS simulates RBN

Simulation idea #2

\[ W(1) \rightarrow \text{init} \rightarrow W(#) \rightarrow \text{sent} \]

\[ W(#) \rightarrow a_1 \rightarrow R(1) \rightarrow W(#) \]

\[ W(2) \rightarrow a_2 \rightarrow R(1) \rightarrow W(#) \]

\[ W(3) \rightarrow a_3 \rightarrow R(2) \rightarrow W(#) \]

\[ b_1 \rightarrow R(2) \rightarrow W(#) \]

\[ b_2 \rightarrow R(2) \rightarrow W(#) \]

\[ b_3 \rightarrow W(3) \rightarrow W(#) \]

\[ \text{final} \]

\[ # \]

RBN and ASMS are Equivalent, C. Weil-Kennedy
RBN simulates ASMS

Simulation of polynomial-size with bijection between configurations

\[ W(1) \rightarrow R(1) \rightarrow W(2) \rightarrow R(2) \rightarrow \text{final} \]
RBN simulates ASMS

Simulation idea #1
RBN simulates ASMS

Simulation idea #1

RBN and ASMS are Equivalent, C. Weil-Kennedy
RBN simulates ASMS

Simulate idea #1

RBN and ASMS are Equivalent, C. Weil-Kennedy
RBN simulates ASMS

Simulation idea #2
RBN simulates ASMS

Simulation idea #2

RBN and ASMS are Equivalent, C. Weil-Kennedy
RBN and ASMS are Equivalent, C. Weil-Kennedy

RBN simulates ASMS

Simulation idea #2
RBN simulates ASMS

Simulation idea #2
RBN and ASMS are Equivalent, C. Weil-Kennedy

RBN simulates ASMS

Simulation idea #2
RBN simulates ASMS

Simulation idea #2

RBN and ASMS are Equivalent, C. Weil-Kennedy
RBN simulates ASMS

Simulation idea #2

![Diagram of RBN and ASMS simulation](attachment:diagram.png)
A **cube** is a boolean combination of constraints

\[ a \leq \#q \leq b \]

\[ \in \mathbb{N} \quad \in \mathbb{N} \cup \infty \]
A **cube** is a boolean combination of constraints $a \leq \#q \leq b \in \mathbb{N}$ or $\in \mathbb{N} \cup \infty$.

**cube-reachability**: given cubes $\mathcal{C}$ and $\mathcal{C}'$, does there exist $M \in \mathcal{C}$ and $M' \in \mathcal{C}'$ such that $M$ reaches $M'$?
RBN and ASMS are polynomial-time equivalent w.r.t. to cube-reachability.

**cube-reachability:** given cubes $\mathcal{C}$ and $\mathcal{C}'$, does there exist $M \in \mathcal{C}$ and $M' \in \mathcal{C}'$ such that $M$ reaches $M'$?
Transfer of results

Unbounded initial cube reachability is PSPACE-complete for RBN

is a combination of constraints of the form

\[ 0 \leq \#q \leq 0 \text{ and } 0 \leq \#q < \infty \]
Transfer of results

Unbounded initial cube reachability is PSPACE-complete for RBN

PSPACE-complete for ASMS

[Delzanno et al., FSTTCS ’12]
Transfer of results

In ASMS, \( (\{k \cdot \text{init}\}, r) \) **almost-surely covers** \textit{final}

if it covers \textit{final} with probability 1 under a uniform stochastic scheduler

[Bouyer et al., ICALP ’16]
Transfer of results

In ASMS, \((\{k \cdot \text{init}\}, r)\) almost-surely covers \textit{final} if it covers \textit{final} with probability 1 under a uniform stochastic scheduler

- \(k \geq 1\) is a \textbf{positive cut-off} if \((\{h \cdot \text{init}\}, r)\) almost-surely covers \textit{final} for all \(h \geq k\)
- \(k \geq 1\) is a \textbf{negative cut-off} if \((\{h \cdot \text{init}\}, r)\) does not almost-surely cover \textit{final} for all \(h \geq k\)
Transfer of results

In ASMS, \((\{k \cdot \text{init}\}, r)\) almost-surely covers \(\text{final}\)
if it covers \(\text{final}\) with probability 1 under a uniform stochastic scheduler.

- \(k \geq 1\) is a **positive cut-off** if \((\{h \cdot \text{init}\}, r)\) almost-surely covers \(\text{final}\)
  for all \(h \geq k\)
- \(k \geq 1\) is a **negative cut-off** if \((\{h \cdot \text{init}\}, r)\) does not almost-surely
  cover \(\text{final}\) for all \(h \geq k\)

All ASMS have a positive or negative cut-off and this can be decided in **EXPSPACE**
Transfer of results

[Bouyer et al., ICALP ’16]

In ASMS, \((\{k \cdot \text{init}\}, r)\) **almost-surely covers** \(\text{final}\)
if it covers \(\text{final}\) with probability 1 under a uniform stochastic scheduler

- \(k \geq 1\) is a **positive cut-off** if \((\{h \cdot \text{init}\}, r)\) almost-surely covers \(\text{final}\)
  for all \(h \geq k\)
- \(k \geq 1\) is a **negative cut-off** if \((\{h \cdot \text{init}\}, r)\) does not almost-surely
  cover \(\text{final}\) for all \(h \geq k\)

All ASMS have a positive or negative cut-off and this can be decided in **EXPSPACE**

All RBN have a positive or negative cut-off and this can be decided in **EXPSPACE**
Conclusion

Further

We also compare RBN / ASMS to immediate observation nets (IO):

- anonymous, identical processes which can communicate by observation.
- RBN can simulate IO nets, but IO nets cannot simulate RBN
Conclusion

Further

We also compare RBN / ASMS to immediate observation nets (IO):
• anonymous, identical processes which can communicate by observation.
• RBN can simulate IO nets, but IO nets cannot simulate RBN

Future

• Find complexity of cube-reachability problem for RBN / ASMS. We have article in progress: PSPACE-complete
• Look at open problems for RBN / ASMS (like universal reachability)
Conclusion

Further

We also compare RBN / ASMS to immediate observation nets (IO):
• anonymous, identical processes which can communicate by observation.
• RBN can simulate IO nets, but IO nets cannot simulate RBN

Future

• Find complexity of cube-reachability problem for RBN / ASMS. We have article in progress: PSPACE-complete
• Look at open problems for RBN / ASMS (like universal reachability)

Thank you!
A **cube** is a boolean combination of constraints $a \leq \# q \leq b \in \mathbb{N} \cup \{\infty\}$

**Parameterized problems**: verifying predicates using boolean operators and reachability operators $pre^*$ and $post^*$ over cubes

$pre^*(C)$ is the set of markings that can reach $C$

$post^*(C)$ is the set of markings that $C$ can reach

E.g. *reachability from cube $C$ to cube $C'$*: $post^*(C) \cap C' \neq \emptyset$
A **cube** is a boolean combination of constraints

\[ a \leq \#q \leq b \]

\[ \in \mathbb{N} \quad \in \mathbb{N} \cup \{\infty\} \]

**Parameterized problems**: verifying predicates using boolean operators and reachability operators \( pre^* \) and \( post^* \) over cubes

\( pre^*(\mathcal{C}) \) is the set of markings that can reach \( \mathcal{C} \)

\( post^*(\mathcal{C}) \) is the set of markings that \( \mathcal{C} \) can reach

*e.g. almost-sure reachability from cube \( \mathcal{C}_{init} \) to cube \( \mathcal{C}_{final} \)*

\[ post^*(\mathcal{C}_{init}) \subseteq pre^*(\mathcal{C}_{final}) \]

C. Weil-Kennedy, TUM