

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



# Idempotent Algebra and Dynamic Networks

Bertrand Ducourthial

Laboratoire Heudiasyc (UMR UTC-CNRS 7253)  
Computer Science Dpt  
Université de Technologie de Compiègne

22/05/2012



Dyn. Network

Computation

Dyn. network

Problem

Id. Algebra

Algorithm

Operator

Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

Routing

Broadcasting

Dyn. Group

Data collect

Conclusion



# Idempotent algebra and dynamic networks

## ① Computation in Dynamic Network

## ② Using Idempotent Algebra

## ③ r-semi-group

## ④ Applications of r-semi-groups

## ⑤ Conclusion



# Summary

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



## 1 Computation in Dynamic Network

Computation on networks

Dynamic network

Problem to solve

## 2 Using Idempotent Algebra

## 3 r-semi-group

## 4 Applications of r-semi-groups

## 5 Conclusion



# Computation on networks

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group

Failure

r-semi-groups  
Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

Routing

Broadcasting

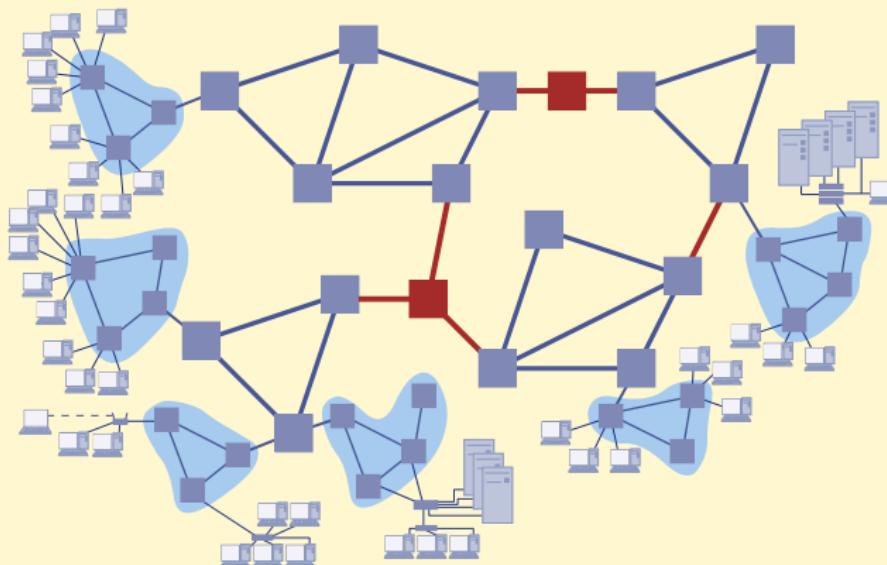
Dyn. Group

Data collect

Conclusion



- Network of communicating computing nodes
  - Local computation using local proc. and memory
  - Local communication with close nodes
  - Each node contributes to a common goal
- Distributed computation



# Dynamic network

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

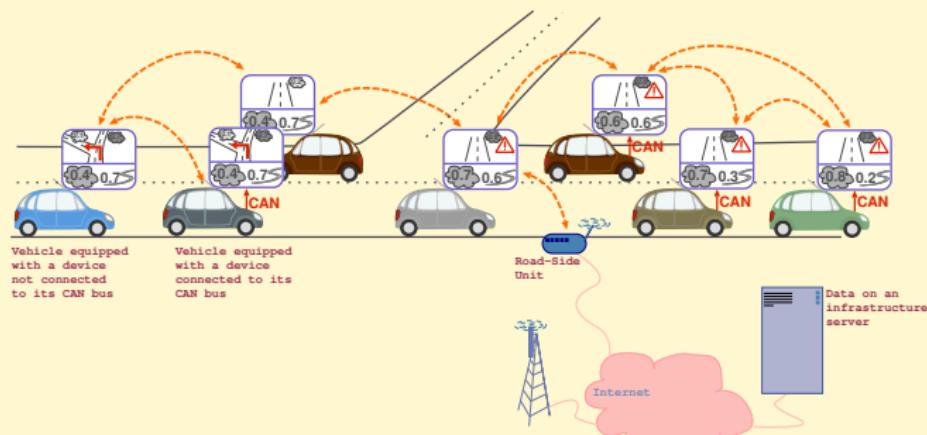
Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect  
Conclusion



- Perturbation of the distributed computation
  - moving nodes ↗ unstable neighborhood
  - node failure
    - permanently, intermittently, unpredict. behavior...
  - link failure
    - messages can appear, disappear or be changed...
  - deliberate attack from malicious agents
- Dynamic network



# Computation on networks

A general problem...

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation

Dyn. network

Problem

Id. Algebra

Algorithm

Operator

Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

Routing

Broadcasting

Dyn. Group

Data collect

Conclusion

**data** → **compute** → **data**



# Computation on networks

A general problem...

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation

Dyn. network

Problem

Id. Algebra

Algorithm

Operator

Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

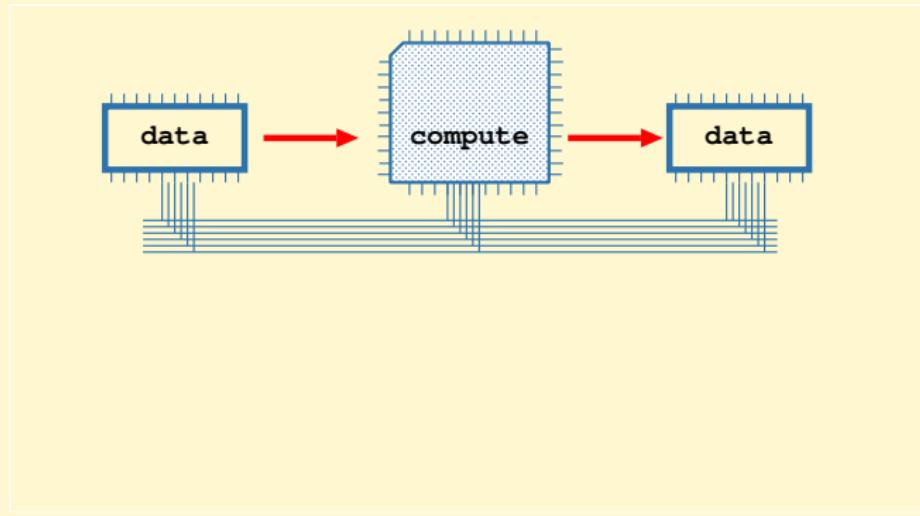
Routing

Broadcasting

Dyn. Group

Data collect

Conclusion



# Computation on networks

A general problem...

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation

Dyn. network

Problem

Id. Algebra

Algorithm

Operator

Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

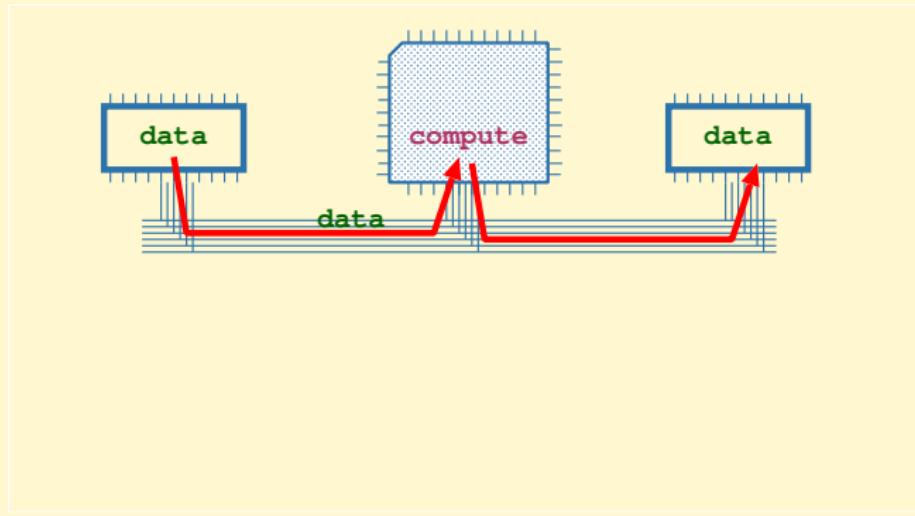
Routing

Broadcasting

Dyn. Group

Data collect

Conclusion



# Computation on networks

A general problem...

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation

Dyn. network

Problem

Id. Algebra

Algorithm

Operator

Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

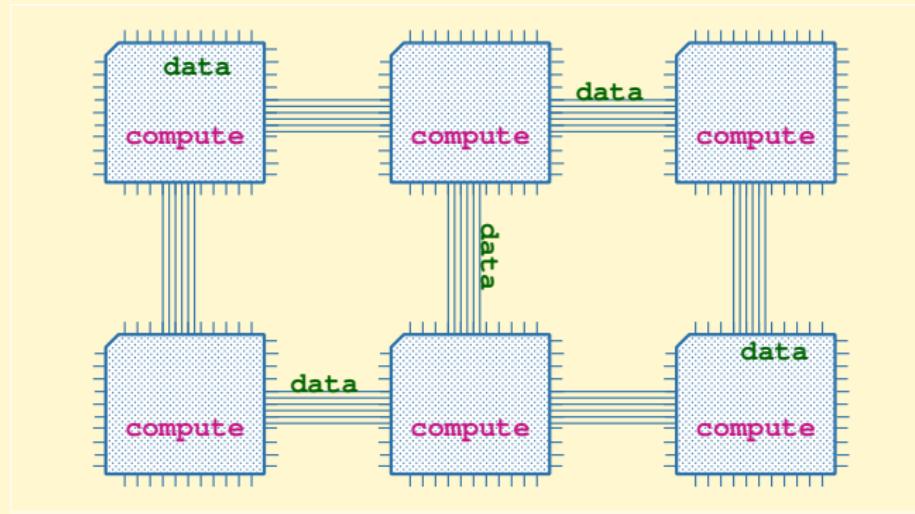
Routing

Broadcasting

Dyn. Group

Data collect

Conclusion



# Computation on networks

A general problem...

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation

Dyn. network

Problem

Id. Algebra

Algorithm

Operator

Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

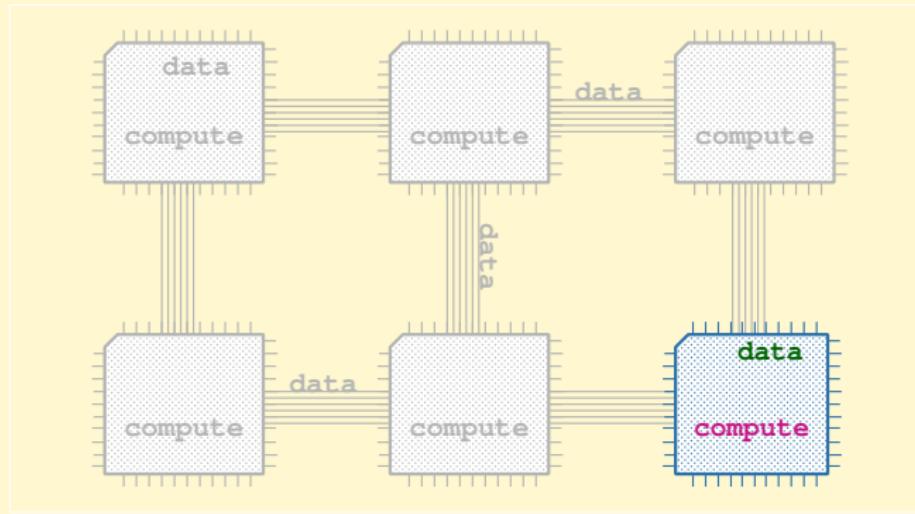
Routing

Broadcasting

Dyn. Group

Data collect

Conclusion

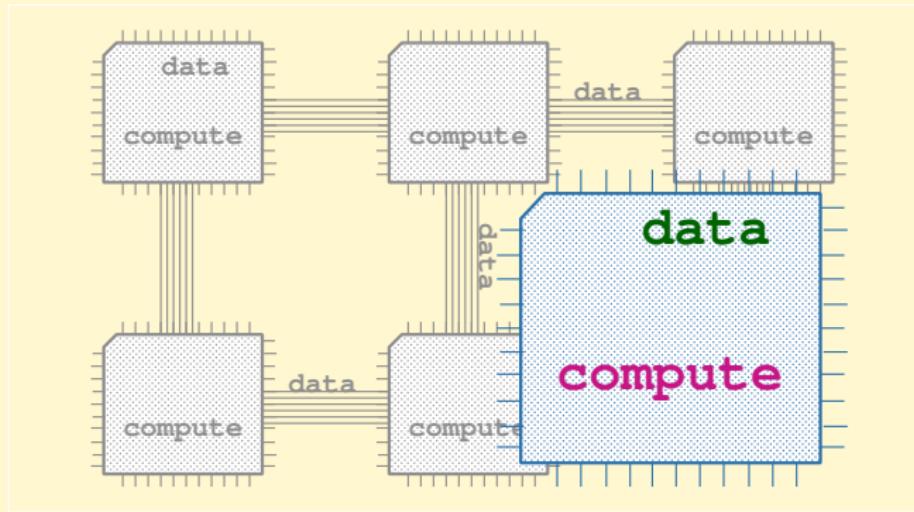


# Computation on networks

## A general problem...

Id. Algebra &  
Dyn. Networks  
  
B. Ducourthial

Dyn. Network  
Computation  
Dyn. network  
Problem  
  
Id. Algebra  
Algorithm  
Operator  
Reminder  
  
r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties  
  
Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect  
  
Conclusion



# Computation on networks

## A general problem...

Id. Algebra &  
Dyn. Networks  
  
B. Ducourthial

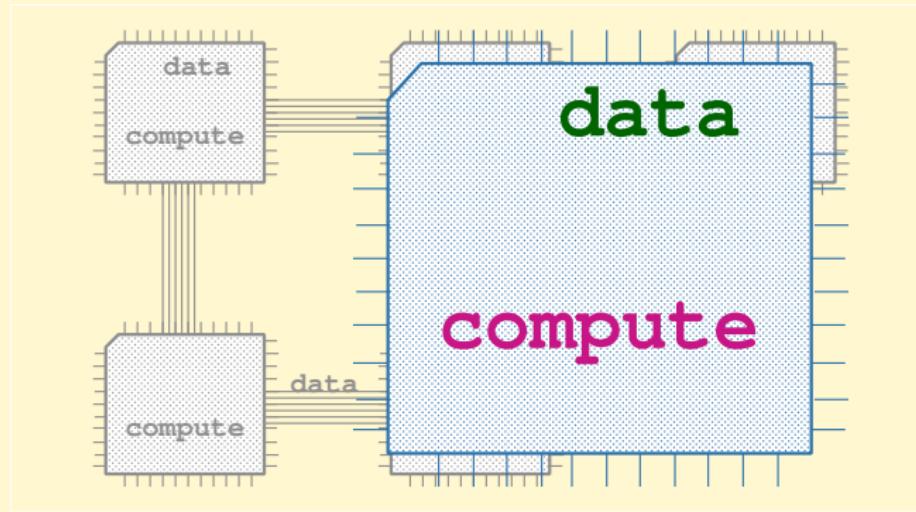
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

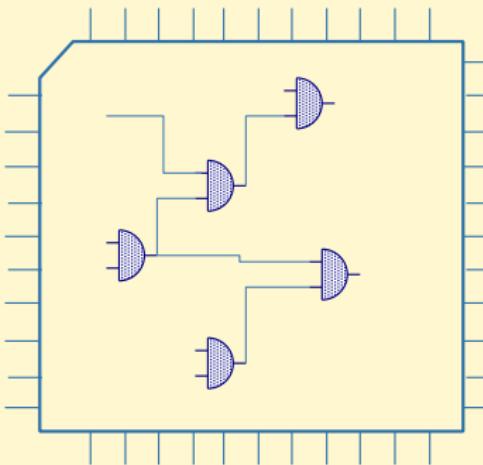
Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



# Computation on networks

A general problem...



Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation

Dyn. network

Problem

Id. Algebra

Algorithm

Operator

Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

Routing

Broadcasting

Dyn. Group

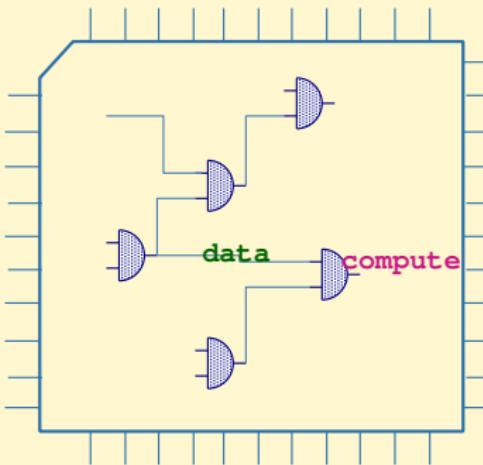
Data collect

Conclusion



# Computation on networks

A general problem...



Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation

Dyn. network

Problem

Id. Algebra

Algorithm

Operator

Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

Routing

Broadcasting

Dyn. Group

Data collect

Conclusion



# Computation on networks

A general problem...

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation

Dyn. network

Problem

Id. Algebra

Algorithm

Operator

Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

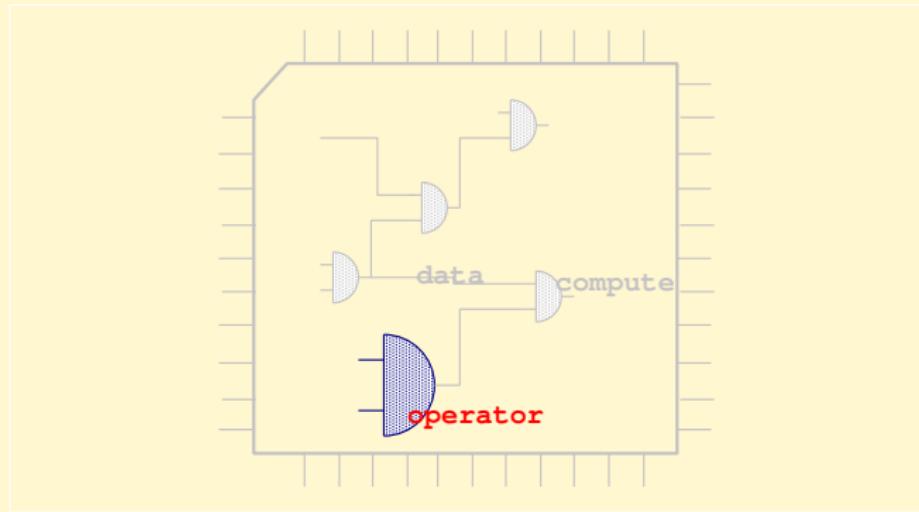
Routing

Broadcasting

Dyn. Group

Data collect

Conclusion



# Computation on networks

## Problem to solve

Id. Algebra &  
Dyn. Networks

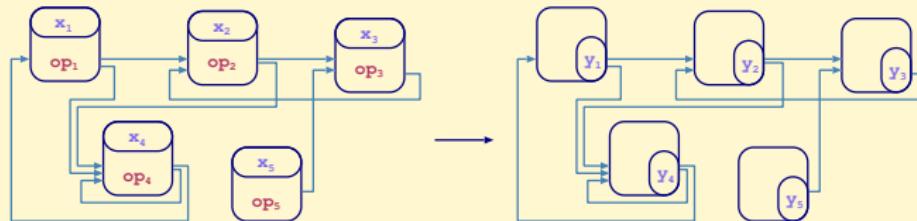
B. Ducourthial

Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect  
Conclusion



- Properties of the operators for
  - termination
  - deterministic result
  - topology awareness
  - stabilization in spite of transient failure
  - with unreliable message passing communication
  - ...and useful for the applications...



# Computation on networks

## Problem to solve

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation

Dyn. network

Problem

Id. Algebra

Algorithm

Operator

Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

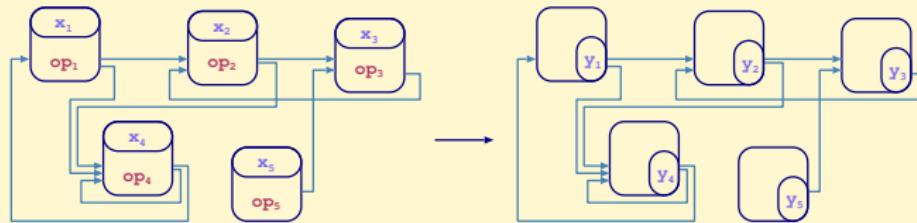
Routing

Broadcasting

Dyn. Group

Data collect

Conclusion



- Properties of the operators for
  - termination
  - deterministic result
  - topology awareness
  - stabilization in spite of transient failure
  - with unreliable message passing communication
  - ...and useful for the applications...

# Summary

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation

Dyn. network

Problem

Id. Algebra

Algorithm

Operator

Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

Routing

Broadcasting

Dyn. Group

Data collect

Conclusion



## 1 Computation in Dynamic Network



## 2 Using Idempotent Algebra

**Algorithm  $\equiv$  operator**

**What kind of operator.**

**A quick reminder on algebra**



## 3 r-semi-group



## 4 Applications of r-semi-groups



## 5 Conclusion



# Local algorithm

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

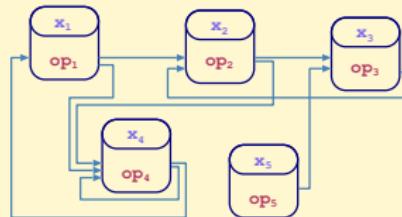
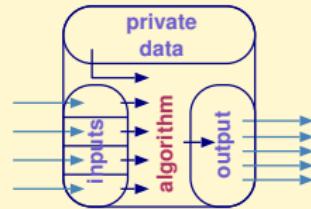
Routing

Broadcasting

Dyn. Group

Data collect

Conclusion



- Local algorithm:

$$\text{output} \leftarrow f(\text{private data}, \text{input}[1], \dots, \text{input}[n])$$

- Distributed algorithm  $\leadsto$  operator(s)



# What kind of operators?

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

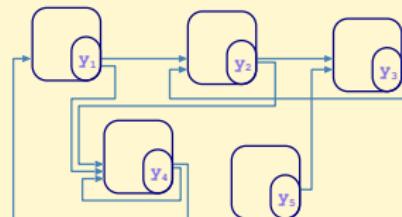
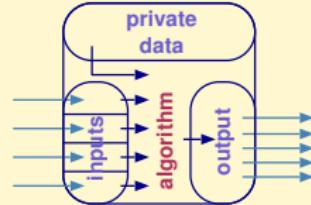
Routing

Broadcasting

Dyn. Group

Data collect

Conclusion



- Static task  $\rightsquigarrow$  stabilization of each output
- Idempotency: stabilization on cycles  
 $x \diamond x = x$
- Commutativity: no computation order  
 $x \diamond y = y \diamond x$
- Associativity: no intermediate computation order  
 $x \diamond (y \diamond z) = (x \diamond y) \diamond z = x \diamond y \diamond z$
- Idempotent Abelian semi-groups  
 $\wedge, \vee, \min, \max, \gcd, \lcm, \cap, \cup \dots$



# A quick reminder on algebra

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

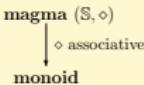
Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion

**magma** ( $\mathbb{S}, \diamond$ )



# A quick reminder on algebra



Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation

Dyn. network

Problem

Id. Algebra

Algorithm

Operator

**Reminder**

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

Routing

Broadcasting

Dyn. Group

Data collect

Conclusion



# A quick reminder on algebra

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation

Dyn. network

Problem

Id. Algebra

Algorithm

Operator

Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

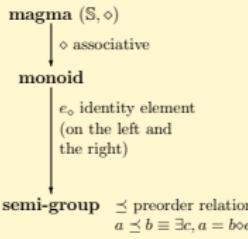
Routing

Broadcasting

Dyn. Group

Data collect

Conclusion



# A quick reminder on algebra

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation

Dyn. network

Problem

Id. Algebra

Algorithm

Operator

Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

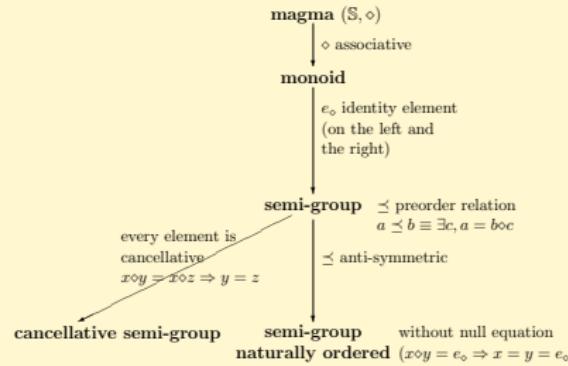
Routing

Broadcasting

Dyn. Group

Data collect

Conclusion



# A quick reminder on algebra

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation

Dyn. network

Problem

Id. Algebra

Algorithm

Operator

Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

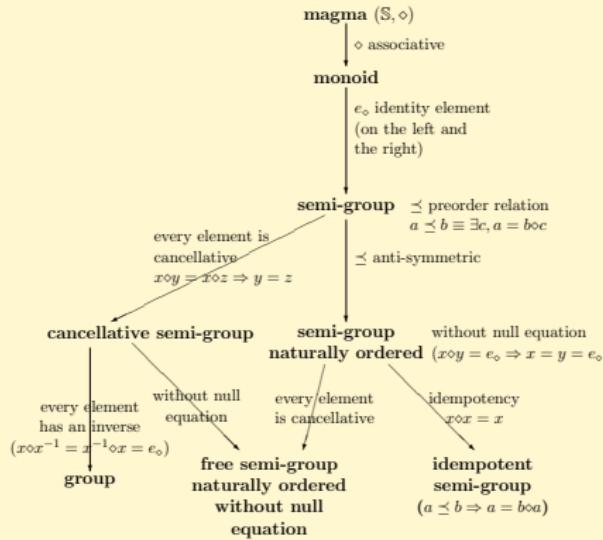
Routing

Broadcasting

Dyn. Group

Data collect

Conclusion



Classical Algebra   Language Algebra   Idempotent Algebra



# A quick reminder on algebra

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation

Dyn. network

Problem

Id. Algebra

Algorithm

Operator

Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

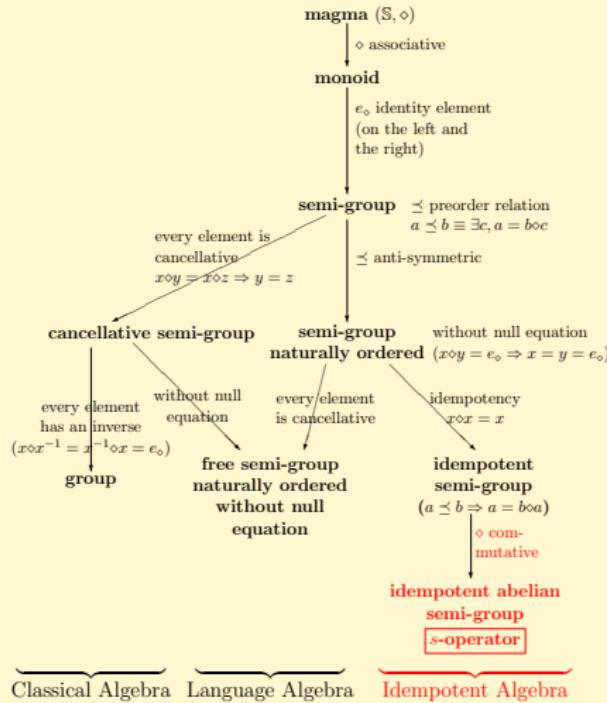
Routing

Broadcasting

Dyn. Group

Data collect

Conclusion



## 1 Computation in Dynamic Network

## 2 Using Idempotent Algebra

## 3 r-semi-group

**Operator versus failure**

**r-semi-groups**

**Prerequisites**

**Definition**

**Properties of r-semi-group**

## 4 Applications of r-semi-groups

## 5 Conclusion



# r-semi-groups: why?

## Operator versus failure

- What happens in case of failure?



# r-semi-groups: why?

## Operator versus failure

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

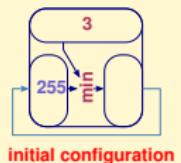
r-semi-group

Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications

Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- What happens in case of failure?



# r-semi-groups: why?

## Operator versus failure

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

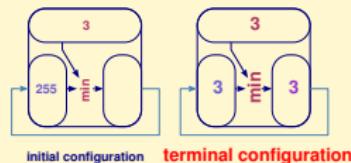
r-semi-group

Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications

Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



# r-semi-groups: why?

## Operator versus failure

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

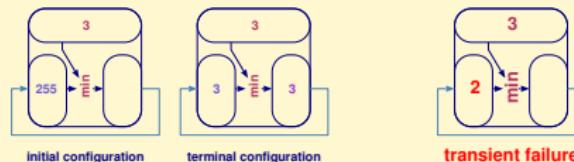
r-semi-group

Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications

Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



# r-semi-groups: why?

## Operator versus failure

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group

Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications

Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



# r-semi-groups: why?

## Operator versus failure

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group

Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

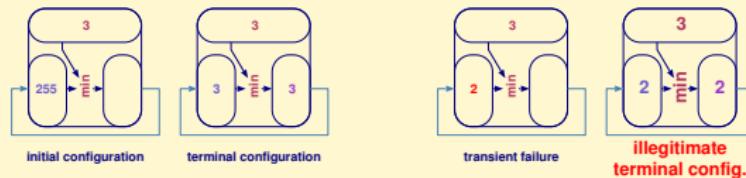
Applications

Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- What happens in case of failure?



- Some operators tolerate failures

- $\text{minc}(x, y) = \min(x, y + 1)$   
defined on  $\mathbb{N} \cup \{+\infty\}$  or  $\{0, \dots, 255\} \dots$

# r-semi-groups: why?

## Operator versus failure

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group

Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

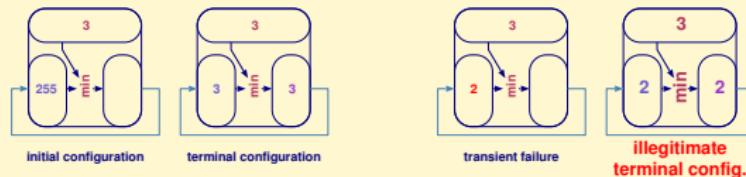
Applications

Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion

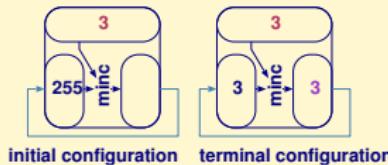


- What happens in case of failure?



- Some operators tolerate failures

- $\text{minc}(x, y) = \min(x, y + 1)$   
defined on  $\mathbb{N} \cup \{+\infty\}$  or  $\{0, \dots, 255\} \dots$



# r-semi-groups: why?

## Operator versus failure

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group

Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

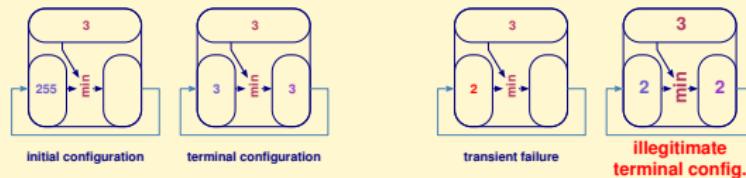
Applications

Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion

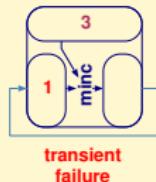
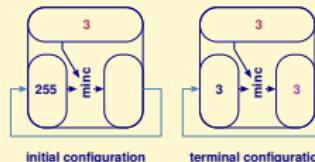


- What happens in case of failure?



- Some operators tolerate failures

- $\text{minc}(x, y) = \min(x, y + 1)$   
 defined on  $\mathbb{N} \cup \{+\infty\}$  or  $\{0, \dots, 255\} \dots$



# r-semi-groups: why?

## Operator versus failure

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group

Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

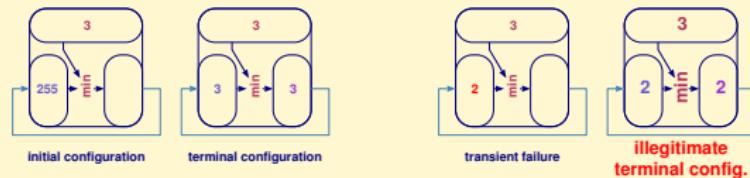
Applications

Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion

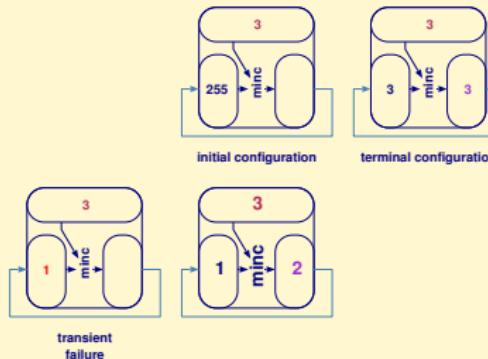


- What happens in case of failure?



- Some operators tolerate failures

- $\text{minc}(x, y) = \min(x, y + 1)$   
 defined on  $\mathbb{N} \cup \{+\infty\}$  or  $\{0, \dots, 255\} \dots$



# r-semi-groups: why?

## Operator versus failure

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group

Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

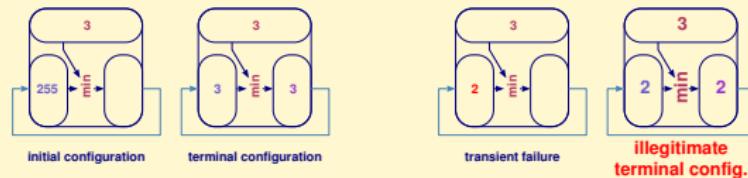
Applications

Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion

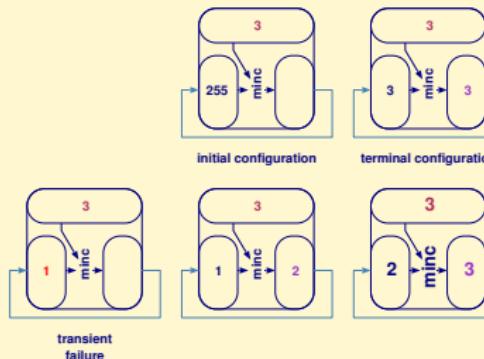


- What happens in case of failure?



- Some operators tolerate failures

- $\text{minc}(x, y) = \min(x, y + 1)$   
 defined on  $\mathbb{N} \cup \{+\infty\}$  or  $\{0, \dots, 255\} \dots$



# r-semi-groups: why?

## Operator versus failure

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group

Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

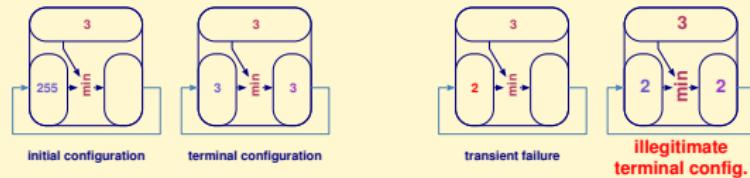
Applications

Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion

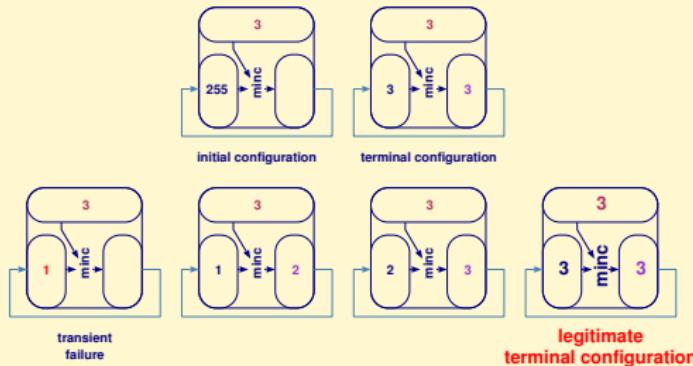


- What happens in case of failure?

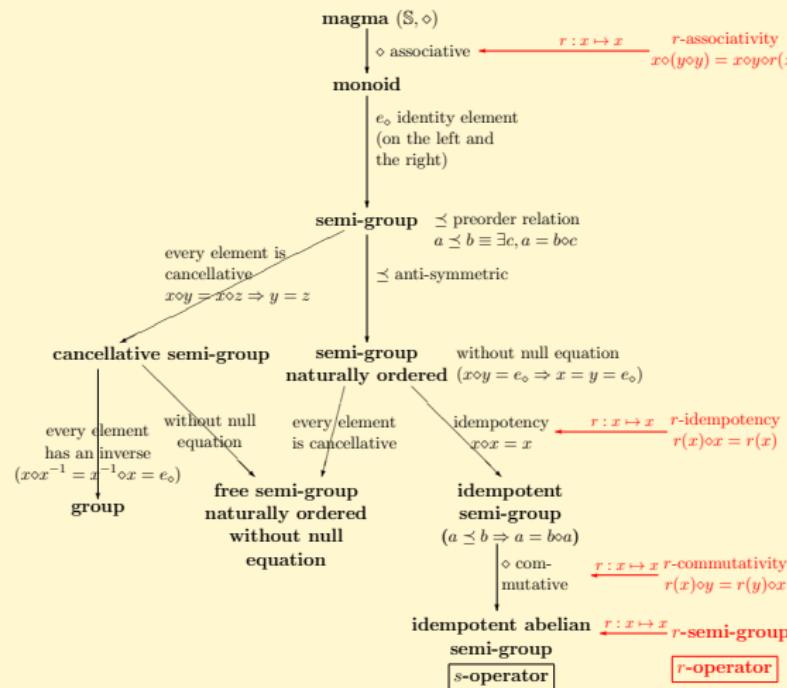


- Some operators tolerate failures

- $\text{minc}(x, y) = \min(x, y + 1)$   
 defined on  $\mathbb{N} \cup \{+\infty\}$  or  $\{0, \dots, 255\} \dots$



# r-semi-groups: what is it?



# r-semi-groups: construction (1)

## Required properties

### • Cancellation

- idempotency vs. cancellation
- $\forall x, y, z \in \mathbb{S}, (y = z) \Rightarrow (x \diamond y = x \diamond z)$   
always true
- $\forall x, y, z \in \mathbb{S}, (x \diamond y = x \diamond z) \Rightarrow (y = z)$   
cancellation  
false for min on  $\mathbb{N}$  (consider  $x = 2, y = 3, z = 4$ )

- $\forall y, z \in \mathbb{S}, (\forall x \in \mathbb{S}, x \diamond y = x \diamond z) \Rightarrow y = z$   
true for min on  $\mathbb{N}$

↝ weak left cancellation  
min is weak left cancellative



# r-semi-groups: construction (1)

## Required properties

### • Cancellation

- idempotency vs. cancellation
- $\forall x, y, z \in \mathbb{S}, (y = z) \Rightarrow (x \diamond y = x \diamond z)$   
always true
- $\forall x, y, z \in \mathbb{S}, (x \diamond y = x \diamond z) \Rightarrow (y = z)$   
cancellation  
false for min on  $\mathbb{N}$  (consider  $x = 2, y = 3, z = 4$ )

- $\forall y, z \in \mathbb{S}, (\forall x \in \mathbb{S}, x \diamond y = x \diamond z) \Rightarrow y = z$   
true for min on  $\mathbb{N}$

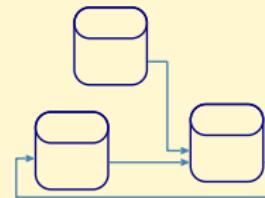
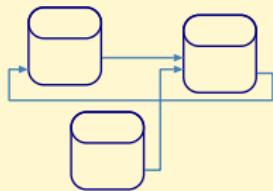
~ $\rightarrow$  **weak left cancellation**  
min is weak left cancellative



# r-semi-groups: construction (1)

## Required properties

- Cancellation ~ weak left cancellation
- Local topology awareness



private data  $\triangleleft$  input[1]  $\triangleleft$  input[2] =  
 private data  $\triangleleft$  input[2]  $\triangleleft$  input[1]

$\sim$  rank 2 commutativity

min is rank 1 commutative (commutative)



# r-semi-groups: construction (1)

## Required properties

- Cancellation ~> weak left cancellation
- Local topology awareness ~> rank commutativity
- Termination

removing doubles in expressions:

private data  $\triangleleft$  input  $\triangleleft$  input =  
 private data  $\triangleleft$  input

~> rank 2 idempotency

min is rank 1 idempotent (idempotent)



# r-semi-groups: construction (2)

Generalization of the semi-group

- **Idempotent Abelian semi-group** e.g., min  
 $(\mathbb{S}, \oplus)$  magma

associative	$(x \oplus y) \oplus z = x \oplus (y \oplus z)$
commutative	$x \oplus y = y \oplus x$
idempotent	$x \oplus x = x$
identity element	$x \oplus e_{\oplus} = x$



Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra

Algorithm  
Operator  
Reminder

r-semi-group

Failure  
r-semi-groups

Prerequisites

Definition  
Properties

Applications

Distance  
Best paths  
Depth first

Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



# r-semi-groups: construction (2)

## Generalization of the semi-group

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation

Dyn. network

Problem

Id. Algebra

Algorithm

Operator

Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

Routing

Broadcasting

Dyn. Group

Data collect

Conclusion



- **Idempotent Abelian semi-group** e.g., min  
 $(\mathbb{S}, \oplus)$  magma

associative       $(x \oplus y) \oplus z = x \oplus (y \oplus z)$

commutative       $x \oplus y = y \oplus x$

idempotent       $x \oplus x = x$

identity element       $x \oplus e_{\oplus} = x$

- **r-semi-group** e.g., minc( $x, y$ ) = min( $x, y + 1$ )

$(\mathbb{S}, \triangleleft)$  weak left cancellative magma

$$\forall y, z \in \mathbb{S}, (\forall x \in \mathbb{S}, x \triangleleft y = x \triangleleft z) \Rightarrow y = z$$

$r : \mathbb{S} \rightarrow \mathbb{S}$  endomorphism

$$\forall x, y \in \mathbb{S}, r(x \triangleleft y) = r(x) \triangleleft r(y)$$

**r**-associative       $(x \triangleleft y) \triangleleft r(z) = x \triangleleft (y \triangleleft z)$

**r**-commutative       $r(x) \triangleleft y = r(y) \triangleleft x$

**r**-idempotent       $r(x) \triangleleft x = r(x)$

right identity elt       $x \triangleleft e_{\triangleleft} = x$



# r-semi-groups: construction (2)

## Generalization of the semi-group

- Idempotent Abelian semi-group e.g., min

- **r-semi-group** e.g.,  $\text{minc}(x, y) = \min(x, y + 1)$

- **Idempotent r-semi-group**

- idempotent *r*-operator:

$$\forall x \in \mathbb{S}, \quad x \triangleleft x = x$$

$$\rightsquigarrow x \preceq_{\triangleleft} r(x)$$

useful for termination

- strictly idempotent *r*-operator:

$$x \prec_{\triangleleft} r(x)$$

useful for self-stabilization



# r-semi-groups: properties

Dyn. Network **magma  $(S, \triangleleft)$**

$\triangleleft$  weak left cancellative

$e_r$  right identity element

Computation  $r : S \rightarrow S$

Dyn. network

Problem

Id. Algebra

$\triangleleft r\text{-associative}$

Algorithm

Operator

Reminder

r-semi-group

$\triangleleft r\text{-commutative}$

Failure

r-semi-groups

Prerequisites

Definition

Properties

$\triangleleft r\text{-idempotent}$

Applications

Distance

Best paths

Depth first

Ancestors

Routing

Broadcasting

$r$  endomorphism  
of  $(S, \triangleleft)$

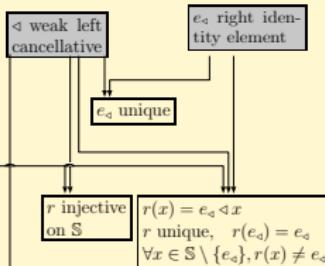
Dyn. Group

Data collect

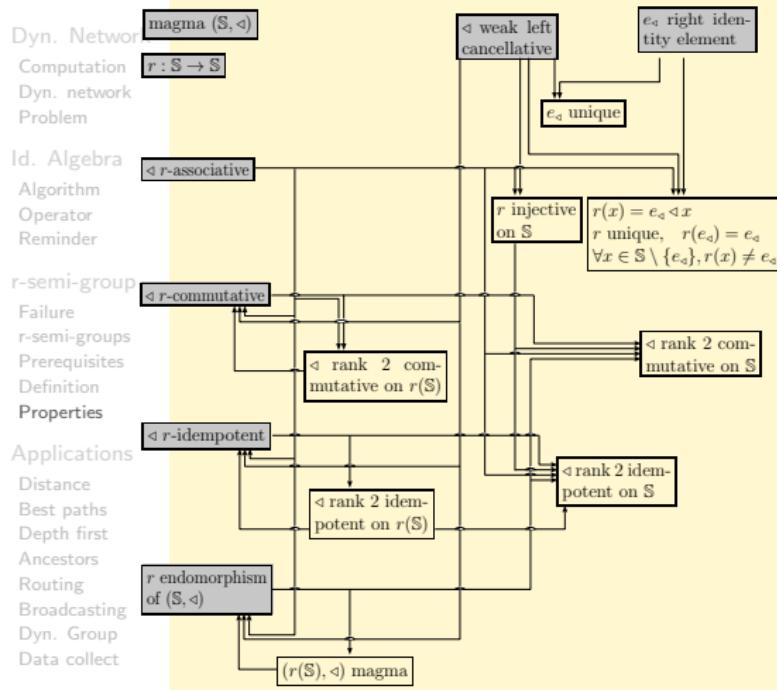
Conclusion



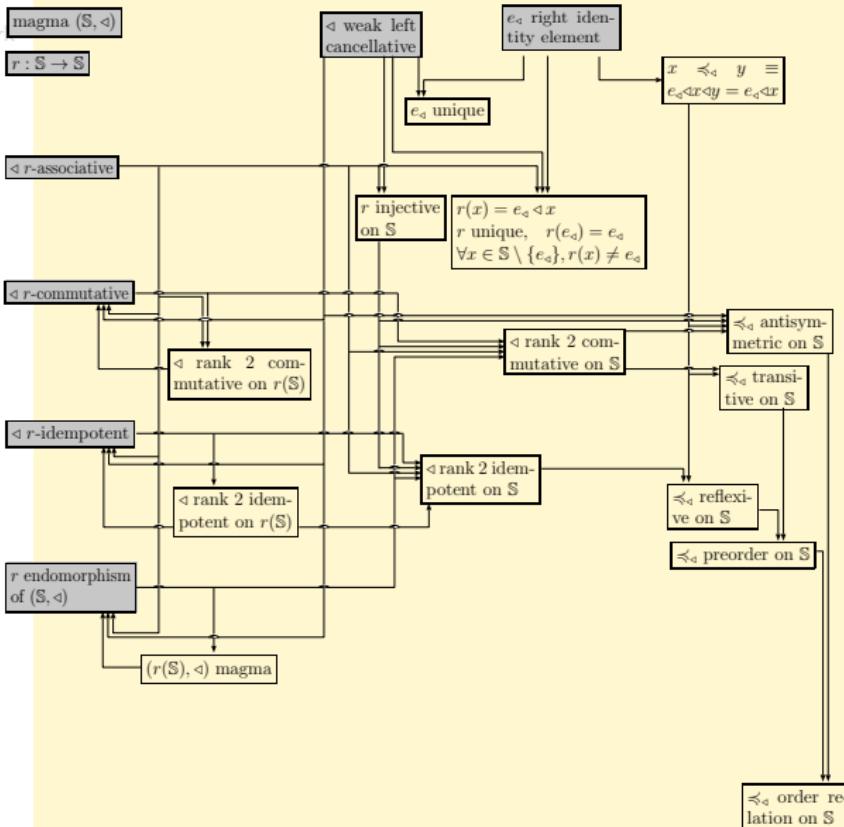
# r-semi-groups: properties



## r-semi-groups: properties



# r-semi-groups: properties



# r-semi-groups: properties

magma  $(S, \triangleleft)$

$\triangleleft$  weak left cancellative

$e_{\triangleleft}$  right identity element

$$\begin{array}{c} x \triangleleft_{\triangleleft} y \equiv \\ e_{\triangleleft} \triangleleft x \triangleleft y = e_{\triangleleft} \triangleleft x \end{array}$$

$\triangleleft$  r-associative

$\triangleleft$  r-commutative

$\triangleleft$  r-idempotent

$r$  endomorphism  
of  $(S, \triangleleft)$

$$\forall x, y \in S, r(x) \triangleleft y \in r(S)$$

$r$  injective  
on  $S$

$$\begin{array}{l} r(x) = e_{\triangleleft} \triangleleft x \\ r \text{ unique}, \quad r(e_{\triangleleft}) = e_{\triangleleft} \\ \forall x \in S \setminus \{e_{\triangleleft}\}, r(x) \neq e_{\triangleleft} \end{array}$$

$\triangleleft_{\triangleleft}$  rank 2 commutative on  $r(S)$

$\triangleleft_{\triangleleft}$  rank 2 idempotent on  $r(S)$

$\triangleleft_{\triangleleft}$  rank 2 commutative on  $r^2(S)$

$\triangleleft_{\triangleleft}$  rank 2 idempotent on  $r^2(S)$

$\triangleleft_{\triangleleft}$  rank 2 commutative on  $r^2(S)$

$\triangleleft_{\triangleleft}$  rank 2 idempotent on  $r^2(S)$

$\triangleleft_{\triangleleft}$  reflexive on  $S$

$\triangleleft_{\triangleleft}$  preorder on  $S$

$\triangleleft_{\triangleleft}$  antisymmetric on  $S$

$\triangleleft_{\triangleleft}$  transitive on  $S$

$$\begin{array}{l} r^2(x) \oplus r^2(y) \equiv \\ r^2(x) \triangleleft r(y) \\ \text{internal composition law} \\ \text{on } r^2(S) \end{array}$$

$\triangleleft_{\triangleleft}$  order relation on  $S$

$\oplus$  associative on  $r^2(S)$

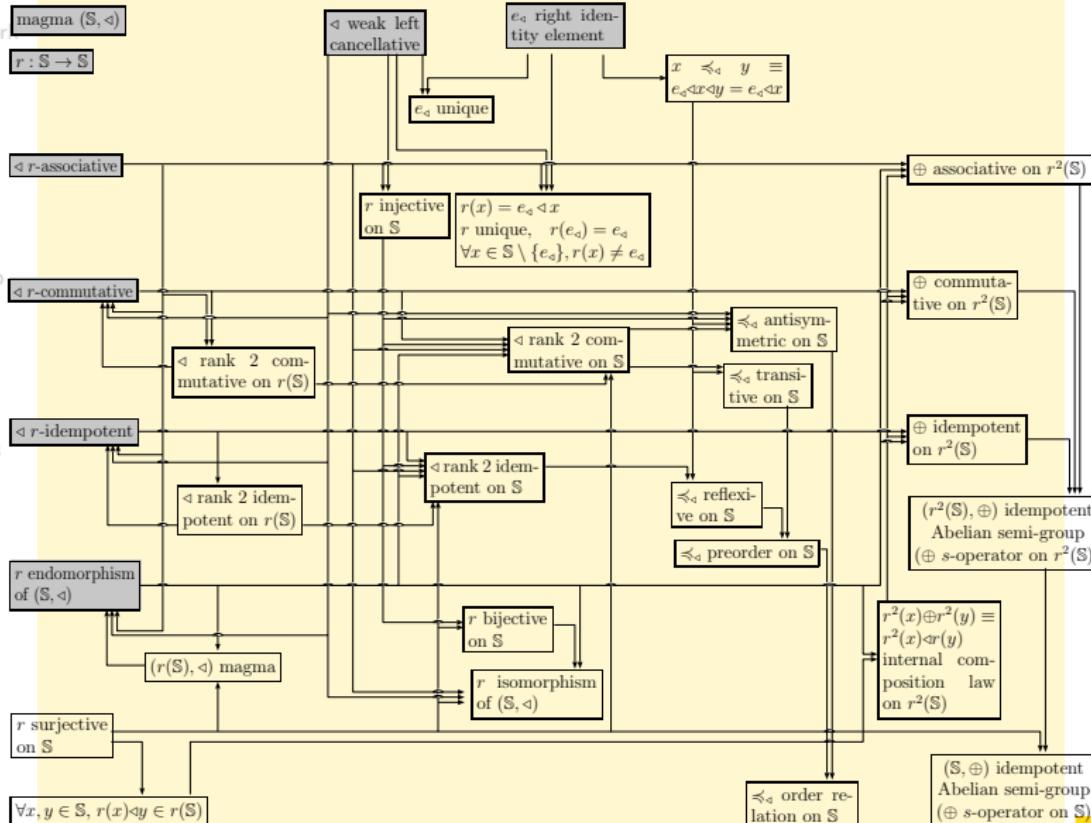
$\oplus$  commutative on  $r^2(S)$

$\oplus$  idempotent on  $r^2(S)$

$(r^2(S), \oplus)$  idempotent  
Abelian semi-group  
 $(\oplus, s\text{-operator on } r^2(S))$



# r-semi-groups: properties



# Summary

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



## 1 Computation in Dynamic Network

## 2 Using Idempotent Algebra

## 3 r-semi-group

## 4 Applications of r-semi-groups

Distance

Best paths

Depth first search tree

Ordered ancestors list

Self-stabilizing routing table construction

Broadcasting

Dynamic group in vehicular networks

Data collect



# Applications of r-semi-groups

## Distance and Shortest paths

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra

Algorithm  
Operator  
Reminder

r-semi-group

Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications

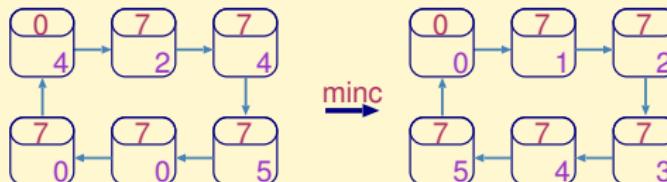
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



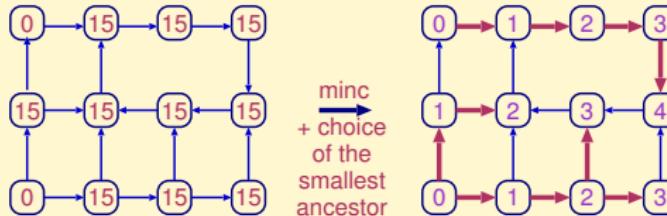
- Distance computation

$$\text{minc}(x, y) = \min(x, y + 1)$$



- Single / Multiple source shortest path

Also with weights on edges



# Applications of r-semi-groups

Best paths

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best reliable paths

- multiplicative criterion  $\pi \in [0, 1]$  on each edge
- $\text{maxmul}(x, y) = \max(x, y \times \pi)$

- Best capacity paths

- path capacity = minimal edge capacity  $\kappa$  along the path
- $\text{maxmin}(x, y) = \max(x, \min(y, \kappa))$



# Applications of r-semi-groups

Depth first search tree

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation

Dyn. network

Problem

Id. Algebra

Algorithm

Operator

Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

Routing

Broadcasting

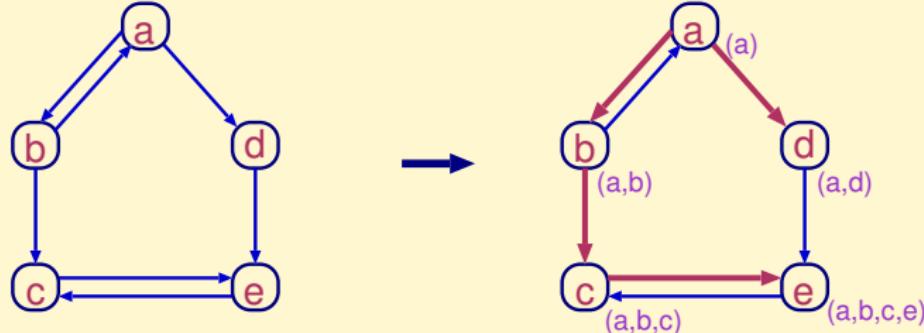
Dyn. Group

Data collect

Conclusion



- semi-group  $(\mathbb{S}, \oplus)$ 
  - lists of nodes
  - $\oplus$ : lexicographic order on  $\mathbb{S}$
  - example:  $(a, b, d) \oplus (a, b, c, d) = (a, b, c, d)$
- $r : \mathbb{S} \rightarrow \mathbb{S}$ 
  - $r(\text{list}) = \text{list} \cup (v)$  on the node  $v$
  - example:  $r(a, b) = (a, b, d)$  on node  $d$
- r-semigroup
  - $x \triangleleft y = x \oplus r(y)$
  - strictly idempotent



# Applications of r-semi-groups

Ordered ancestors list

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation

Dyn. network

Problem

Id. Algebra

Algorithm

Operator

Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

Routing

Broadcasting

Dyn. Group

Data collect

Conclusion



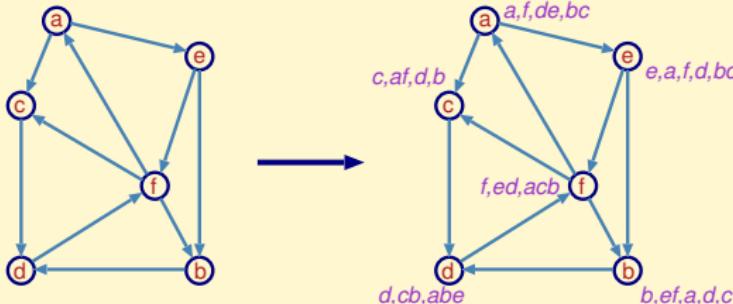
- semi-group  $(\mathbb{S}, \oplus)$

- lists of sets of vertices  $(S_1, \dots, S_k)$
- $\oplus$ : term to term merging + deleting doubles
- ex.:  $(\{d\}, \{b\}, \{a, c\}) \oplus (\{c\}, \{a, e\}, \{b\}) = (\{d, c\}, \{b, a, e\}, \{\text{ }a, \text{ }c, \text{ }b\}) = (\{d, c\}, \{b, a, e\})$

- $r((S_1, \dots, S_k)) = (\emptyset, S_1, \dots, S_k)$

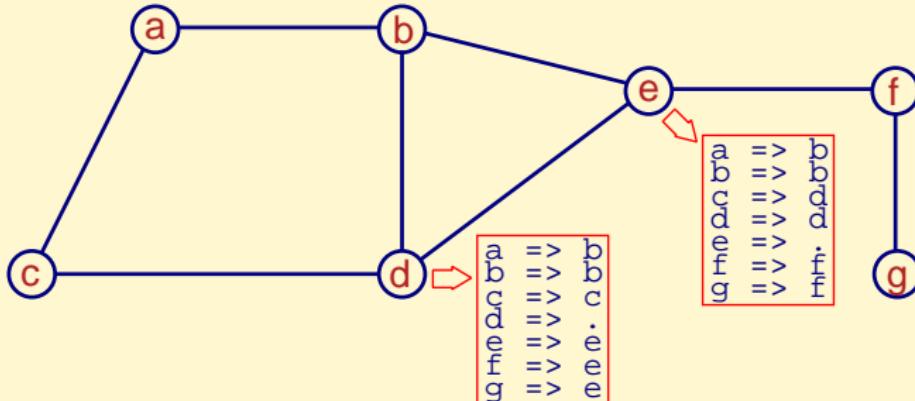
- **r-semigroup**

- $x \triangleleft y = x \oplus r(y)$
- strictly idempotent



# Applications of r-semi-groups

## Routing table construction (1)



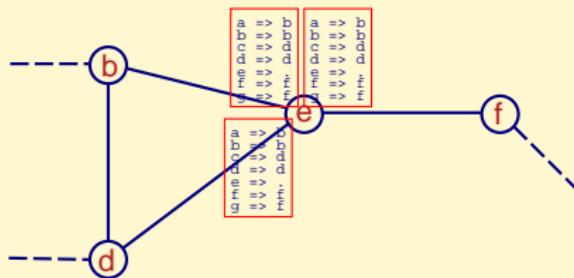
- How to design a self-stabilizing algorithm for routing tables construction?



# Applications of r-semi-groups

## Routing table construction (2)

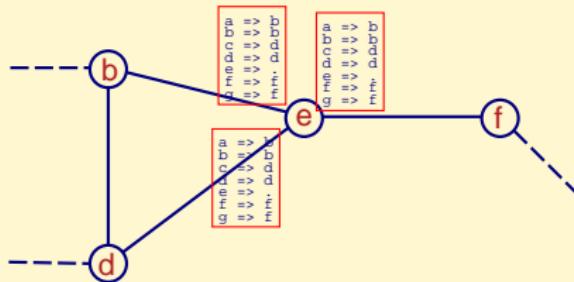
- A node periodically sends its local informations to its neighbors



# Applications of r-semi-groups

## Routing table construction (2)

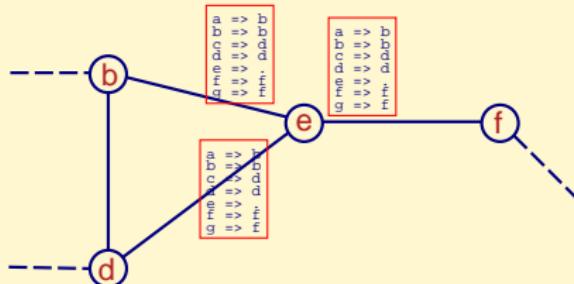
- A node periodically sends its local informations to its neighbors



# Applications of r-semi-groups

## Routing table construction (2)

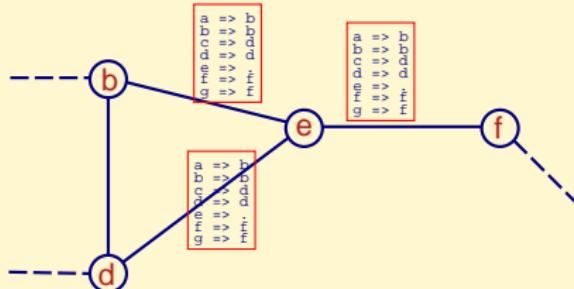
- A node periodically sends its local informations to its neighbors



# Applications of r-semi-groups

## Routing table construction (2)

- A node periodically sends its local informations to its neighbors



Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications

Distance  
Best paths  
Depth first  
Ancestors

**Routing**

Broadcasting  
Dyn. Group  
Data collect

Conclusion



# Applications of r-semi-groups

## Routing table construction (2)

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra

Algorithm  
Operator  
Reminder

r-semi-group

Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications

Distance  
Best paths  
Depth first  
Ancestors

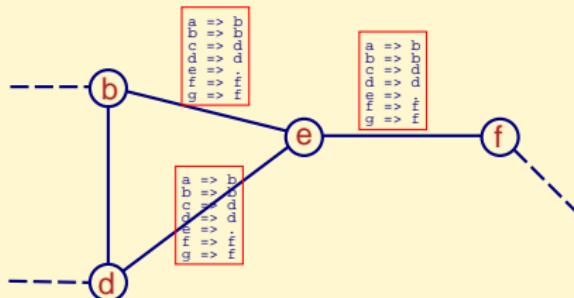
**Routing**

Broadcasting  
Dyn. Group  
Data collect

Conclusion



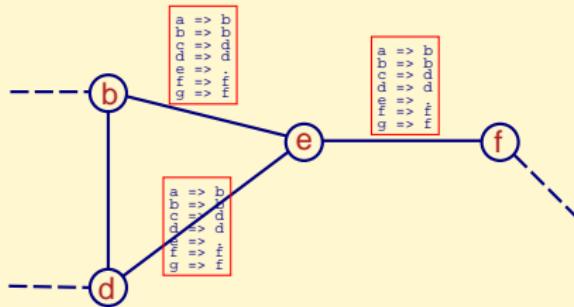
- A node periodically sends its local informations to its neighbors



# Applications of r-semi-groups

## Routing table construction (2)

- A node periodically sends its local informations to its neighbors



# Applications of r-semi-groups

## Routing table construction (3)

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation

Dyn. network

Problem

Id. Algebra

Algorithm

Operator

Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

Routing

Broadcasting

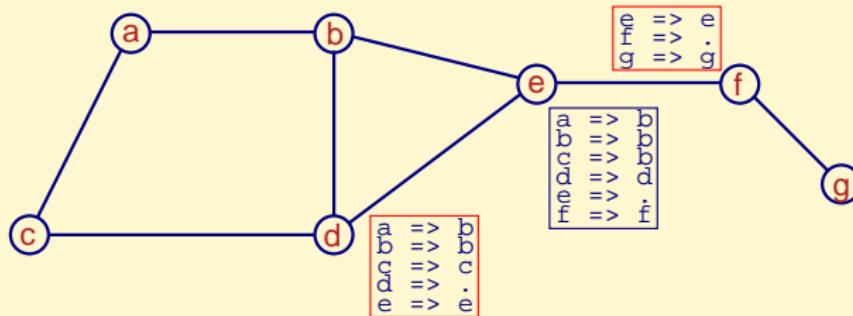
Dyn. Group

Data collect

Conclusion



- When a node receives an information regarding a destination node, it checks:
  - if it has no information for this node  $\leadsto$  add
  - if it has a worse information  $\leadsto$  update



# Applications of r-semi-groups

## Routing table construction (3)

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group

Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

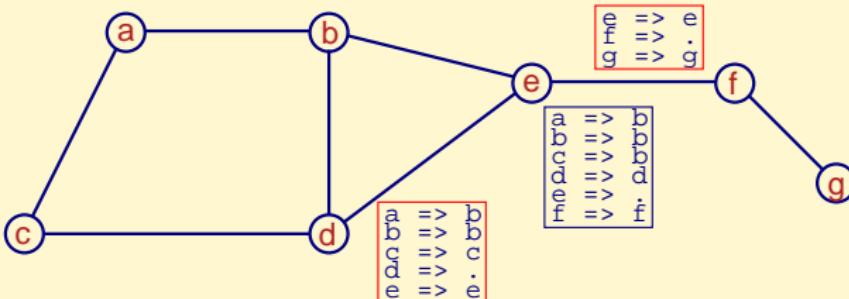
Applications

Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



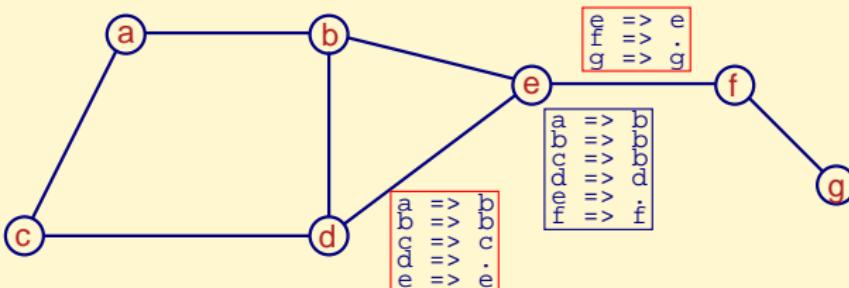
- When a node receives an information regarding a destination node, it checks:
  - if it has no information for this node  $\leadsto$  add
  - if it has a worse information  $\leadsto$  update



# Applications of r-semi-groups

## Routing table construction (3)

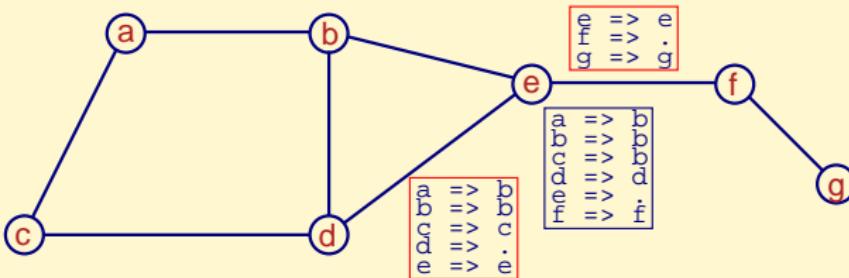
- When a node receives an information regarding a destination node, it checks:
  - if it has no information for this node  $\leadsto$  add
  - if it has a worse information  $\leadsto$  update



# Applications of r-semi-groups

## Routing table construction (3)

- When a node receives an information regarding a destination node, it checks:
  - if it has no information for this node  $\leadsto$  add
  - if it has a worse information  $\leadsto$  update



# Applications of r-semi-groups

## Routing table construction (3)

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group

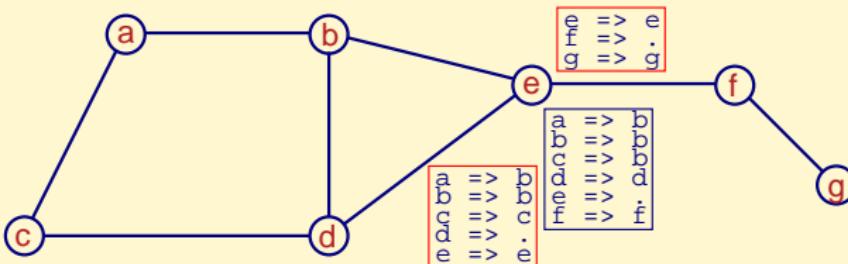
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- When a node receives an information regarding a destination node, it checks:
  - if it has no information for this node  $\leadsto$  add
  - if it has a worse information  $\leadsto$  update



# Applications of r-semi-groups

## Routing table construction (3)

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group

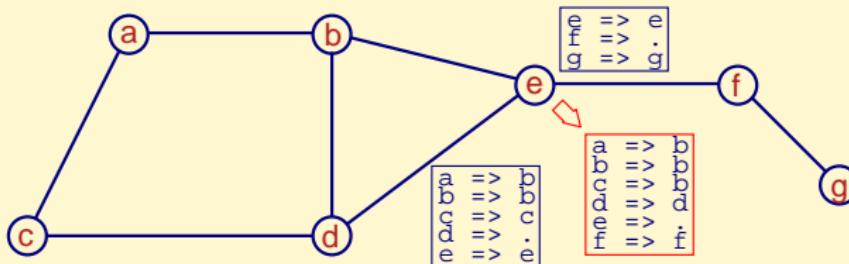
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- When a node receives an information regarding a destination node, it checks:
  - if it has no information for this node  $\leadsto$  add
  - if it has a worse information  $\leadsto$  update



# Applications of r-semi-groups

## Routing table construction (3)

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group

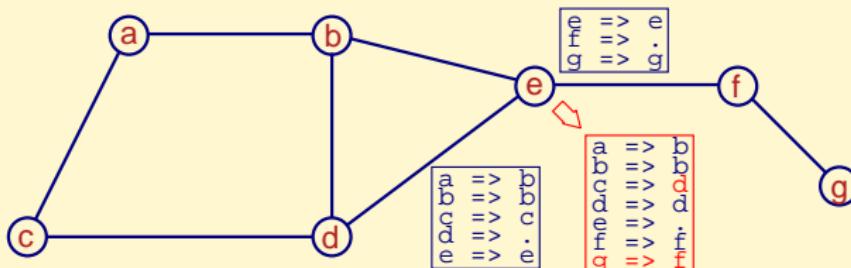
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- When a node receives an information regarding a destination node, it checks:
  - if it has no information for this node  $\leadsto$  add
  - if it has a worse information  $\leadsto$  update

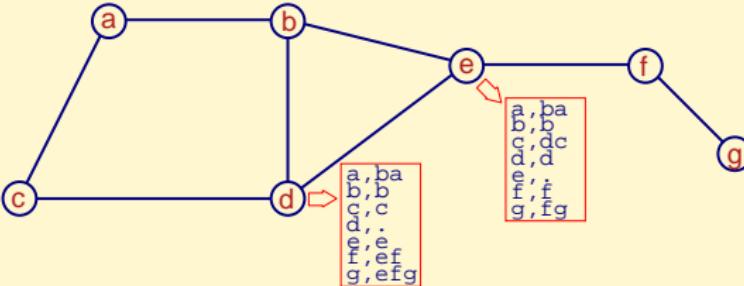


# Applications of r-semi-groups

## Routing table construction (4)

- Data format

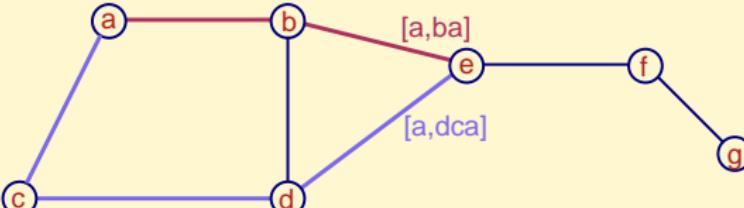
- Information regarding a destination:
  - next hop to reach the destination
  - information regarding the path to select the best next hop
- Entry in the routing table:  
[destination, path to the destination]
- Local information: list of entries  
example:  
 $[a, ba], [b, b], [c, c], [d, \emptyset], [e, e], [f, ef], [g, efg]$



# Applications of r-semi-groups

## Routing table construction (5)

- Choosing the best path to a given destination
  - operator *best* on the paths to a common node  
return (for instance) the shortest path  $\oplus$
  - example (shortest path):  $ba \oplus dca = ba$
  - operator *best* on the similar tables entries  
return the entry with the best path  $\boxplus$
  - example (shortest path):  
 $[a, ba] \boxplus [a, dca] = [a, ba \oplus dca] = [a, ba]$



- $(\mathbb{S}, \boxplus)$  idempotent Abelian semi-group



# Applications of r-semi-groups

## Routing table construction (5)

- Choosing the best path to a given destination
- Building a new table from two tables
  - no more than one path for a destination  
↗ choosing the best
  - operator **fusion** on the list of entries:  $\uplus$ 
    - union of the list
    - choice of the best entry in case of same destination using the operator *best*
  - example:
$$\begin{aligned}
 & ([a, a], [b, b], [c, abc]) \uplus ([a, da], [c, dc], [d, d]) \\
 & = ([a, a] \uplus [a, da], [b, b], [c, abc] \uplus [c, dc], [d, d]) \\
 & = ([a, a \oplus da], [b, b], [c, abc \oplus dc], [d, d])
 \end{aligned}$$
- $(\mathbb{S}, \uplus)$  idempotent Abelian semi-group



# Applications of r-semi-groups

## Routing table construction (5)

- Choosing the best path to a given destination
- Building a new table from two tables
- Properties of the operators
  - operator best on the paths  $\oplus$ 
    - associative  

$$ba \oplus (dca \oplus bcda) = (ba \oplus dca) \oplus bcda$$
    - commutative  $ba \oplus dca = dca \oplus ba$
    - idempotent  $ba \oplus ba = ba$
  - operator best on the entries  $\boxplus \rightsquigarrow$  idem  

$$[a, ba] \boxplus [a, dca] = [a, ba \oplus dca] = [a, ba]$$
  - operator fusion on the lists of entries  $\boxtimes \rightsquigarrow$  idem
- $(\mathbb{S}, \boxtimes)$  idempotent Abelian semi-group



Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group

Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications

Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



# Applications of r-semi-groups

## Routing table construction (5)

- Choosing the best path to a given destination
- Building a new table from two tables
- Properties of the operators
  - operator best on the paths  $\oplus$ 
    - associative  $ba \oplus (dca \oplus bcda) = (ba \oplus dca) \oplus bcda$
    - commutative  $ba \oplus dca = dca \oplus ba$
    - idempotent  $ba \oplus ba = ba$
  - operator best on the entries  $\boxplus \rightsquigarrow$  idem  
 $[a, ba] \boxplus [a, dca] = [a, ba \oplus dca] = [a, ba]$
  - operator fusion on the lists of entries  $\boxtimes \rightsquigarrow$  idem
- $(\mathbb{S}, \boxplus)$  idempotent Abelian semi-group



Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group

Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications

Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



# Applications of r-semi-groups

## Routing table construction (6)

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications

Distance  
Best paths  
Depth first  
Ancestors

Routing

Broadcasting  
Dyn. Group  
Data collect

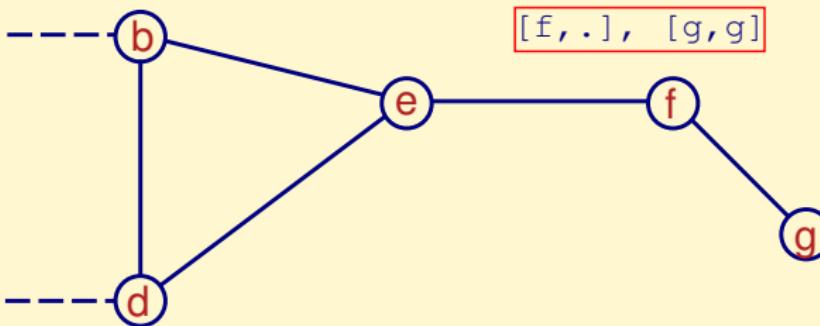
Conclusion



heudiasyc

- ***r***: adding the sender at the begining of each path:

if the node *f* sends the list  $([f, .], [g, g])$ , this list becomes  $([f, f], [g, fg])$  at the arrival



- ***r*-semigroup:**

$$\text{list} \triangleleft \text{list}' = \text{list} \quad \uplus \quad r(\text{list}')$$

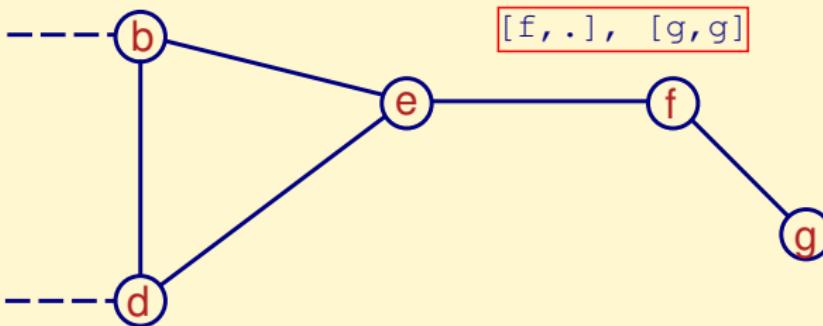


# Applications of r-semi-groups

## Routing table construction (6)

- **r**: adding the sender at the begining of each path:

if the node  $f$  sends the list  $([f, .], [g, g])$ , this list becomes  $([f, f], [g, fg])$  at the arrival



- **r-semigroup**:

$$\text{list} \triangleleft \text{list}' = \text{list} \quad \uplus \quad r(\text{list}')$$



# Applications of r-semi-groups

## Routing table construction (6)

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications

Distance  
Best paths  
Depth first  
Ancestors

Routing

Broadcasting  
Dyn. Group  
Data collect

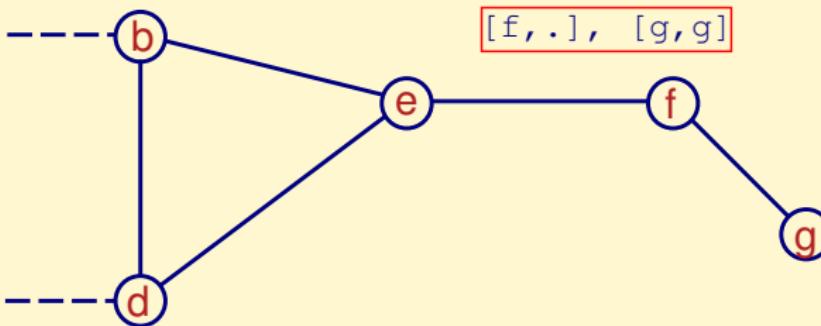
Conclusion



heudiasyc

- ***r***: adding the sender at the begining of each path:

if the node *f* sends the list  $([f, .], [g, g])$ , this list becomes  $([f, f], [g, fg])$  at the arrival



- ***r-semigroup***:

$$\text{list} \triangleleft \text{list}' = \text{list} \quad \uplus \quad r(\text{list}')$$



# Applications of r-semi-groups

## Routing table construction (6)

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications

Distance  
Best paths  
Depth first  
Ancestors

Routing

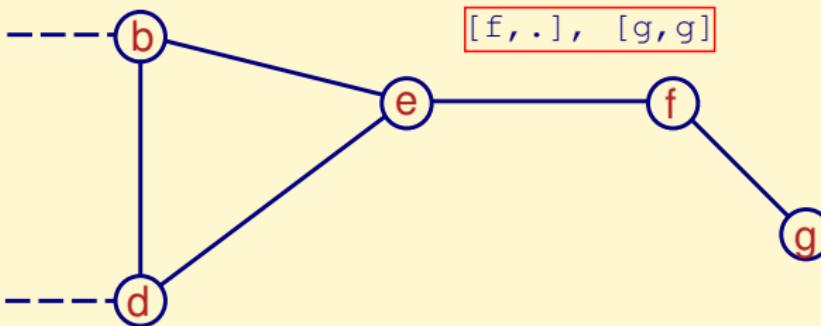
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- ***r***: adding the sender at the begining of each path:

if the node *f* sends the list  $([f, .], [g, g])$ , this list becomes  $([f, f], [g, fg])$  at the arrival



- ***r-semigroup*:**

$$\text{list} \triangleleft \text{list}' = \text{list} \quad \uplus \quad r(\text{list}')$$

# Applications of r-semi-groups

## Routing table construction (6)

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications

Distance  
Best paths  
Depth first  
Ancestors

Routing

Broadcasting  
Dyn. Group  
Data collect

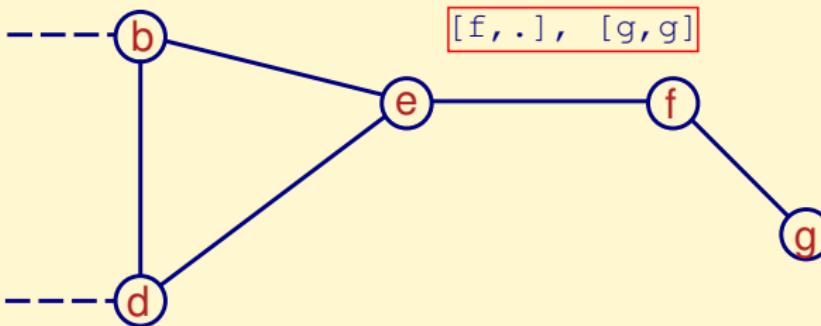
Conclusion



heudiasyc

- ***r***: adding the sender at the begining of each path:

if the node *f* sends the list  $([f, .], [g, g])$ , this list becomes  $([f, f], [g, fg])$  at the arrival



- ***r-semigroup***:

$$\text{list} \triangleleft \text{list}' = \text{list} \quad \uplus \quad r(\text{list}')$$

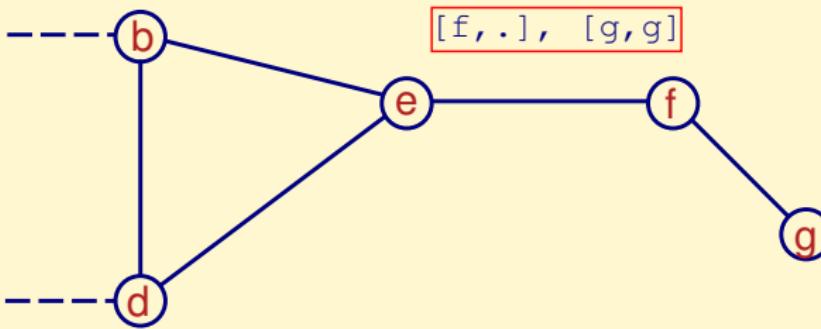


# Applications of r-semi-groups

## Routing table construction (6)

- **r**: adding the sender at the begining of each path:

if the node  $f$  sends the list  $([f, .], [g, g])$ , this list becomes  $([f, f], [g, fg])$  at the arrival



- **r-semigroup**:

$$\text{list} \triangleleft \text{list}' = \text{list} \quad \uplus \quad r(\text{list}')$$

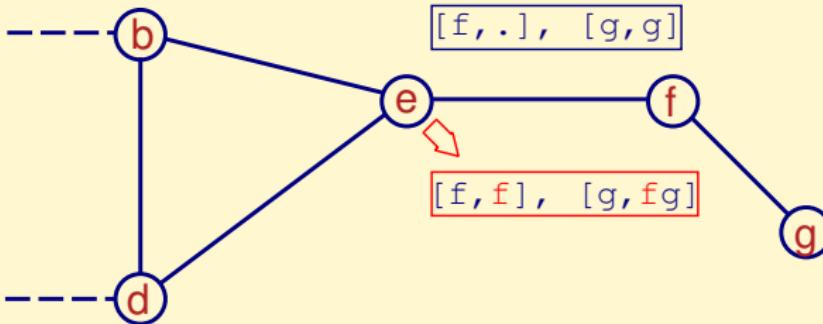


# Applications of r-semi-groups

## Routing table construction (6)

- **r**: adding the sender at the begining of each path:

if the node  $f$  sends the list  $([f, .], [g, g])$ , this list becomes  $([f, f], [g, fg])$  at the arrival



- **r-semigroup:**

$$\text{list} \triangleleft \text{list}' = \text{list} \quad \uplus \quad r(\text{list}')$$



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

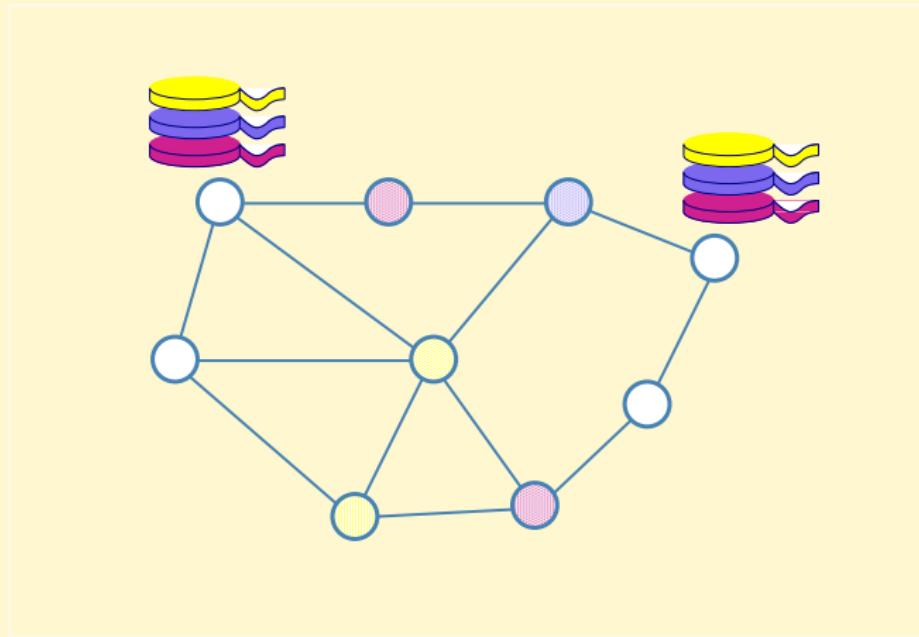
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- **Self-stabilizing distributed algorithm**



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

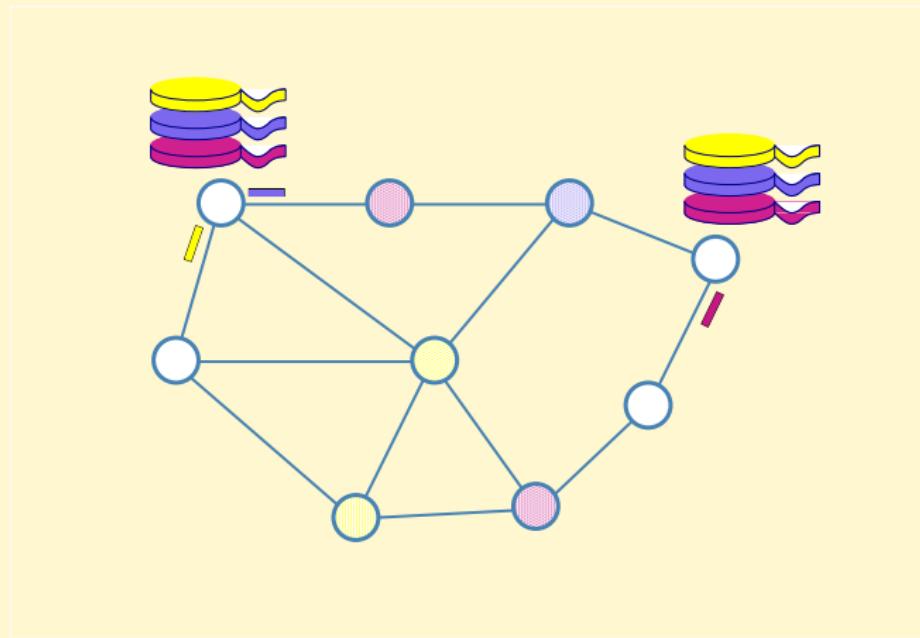
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- **Self-stabilizing distributed algorithm**



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

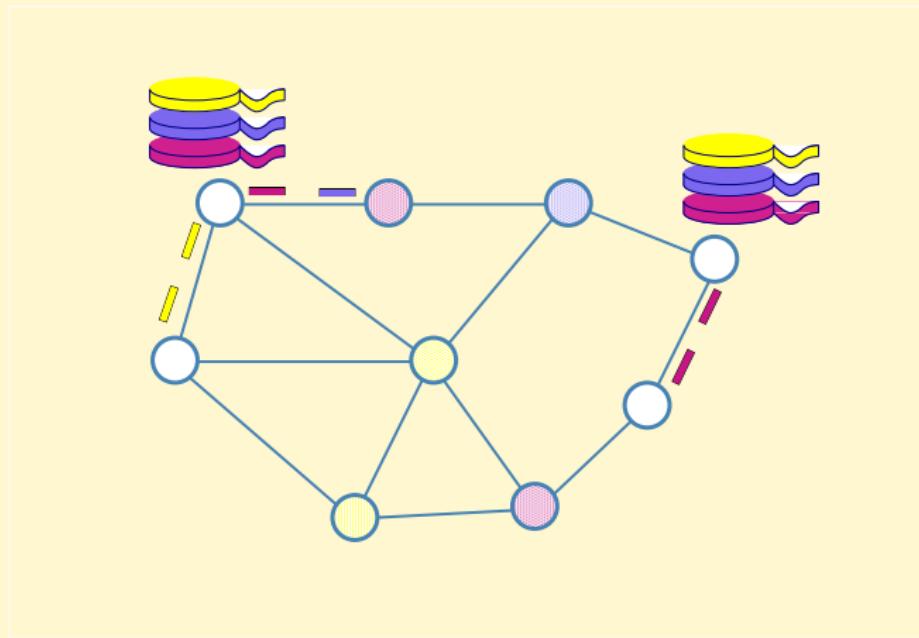
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- **Self-stabilizing distributed algorithm**



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

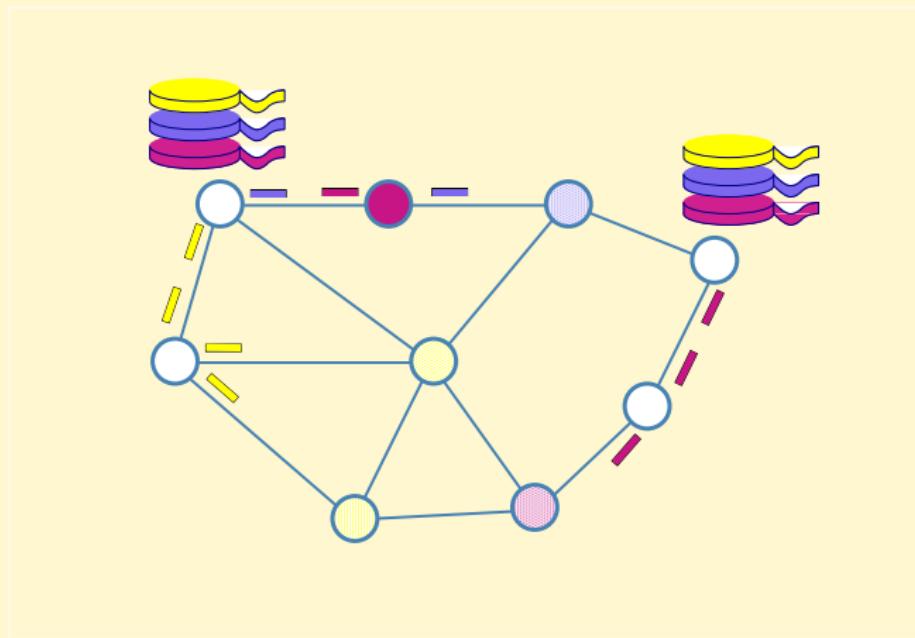
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- **Self-stabilizing distributed algorithm**



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

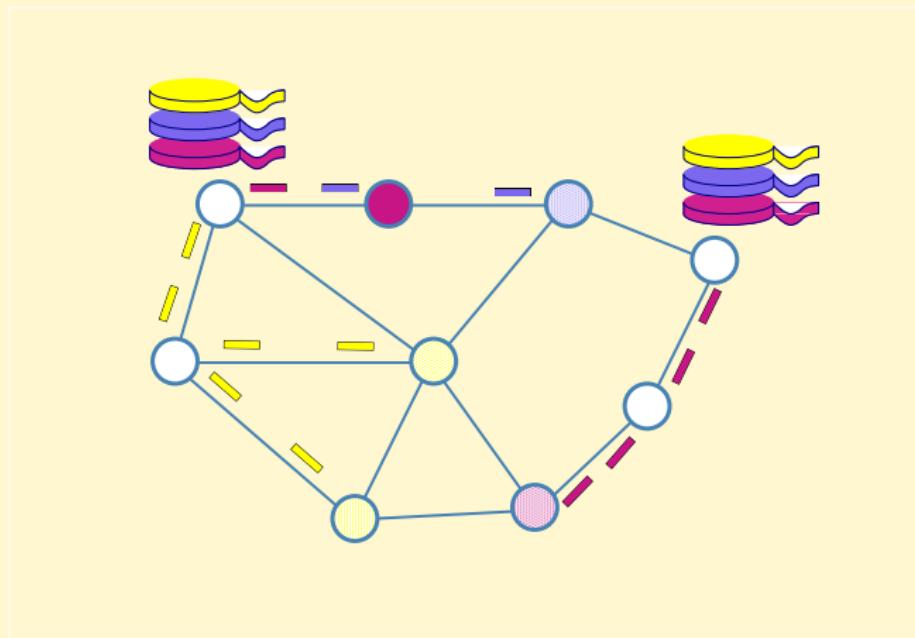
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- **Self-stabilizing distributed algorithm**



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

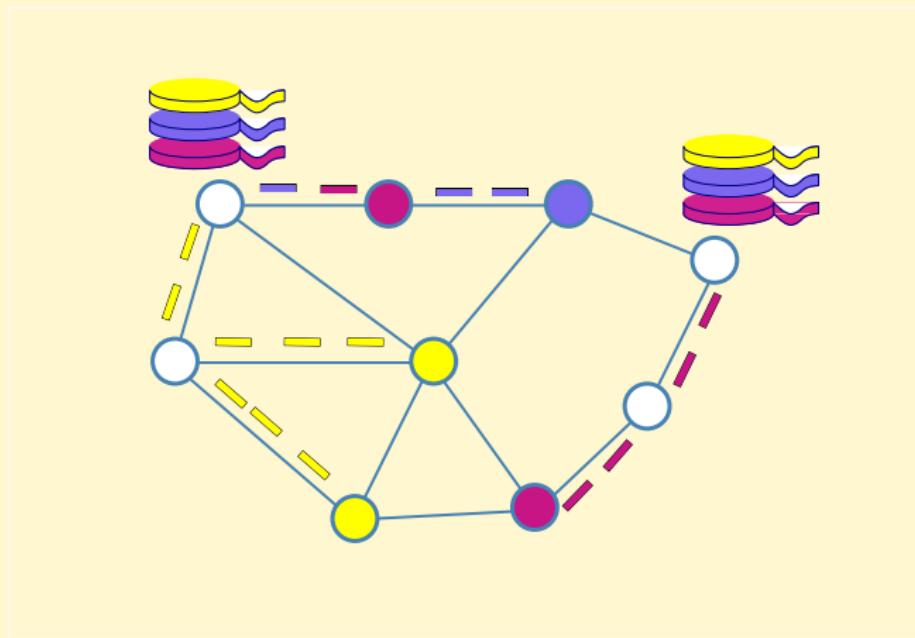
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- Self-stabilizing distributed algorithm



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

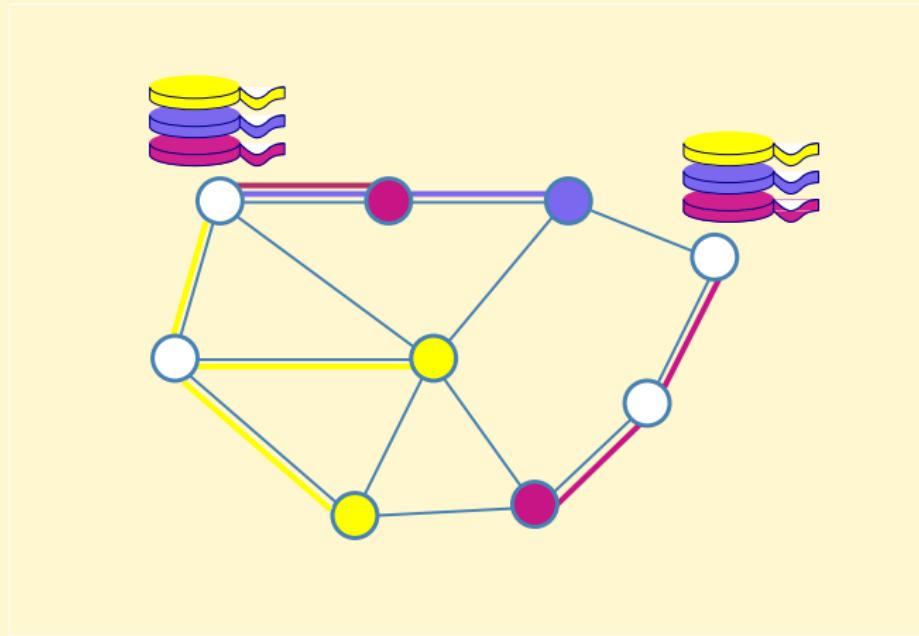
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- Self-stabilizing distributed algorithm



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

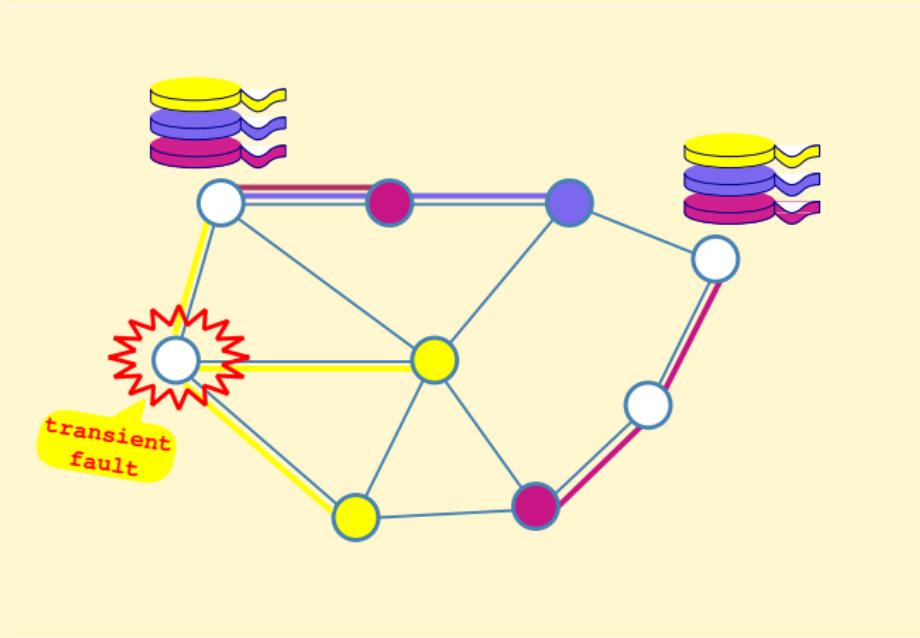
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- Self-stabilizing distributed algorithm



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

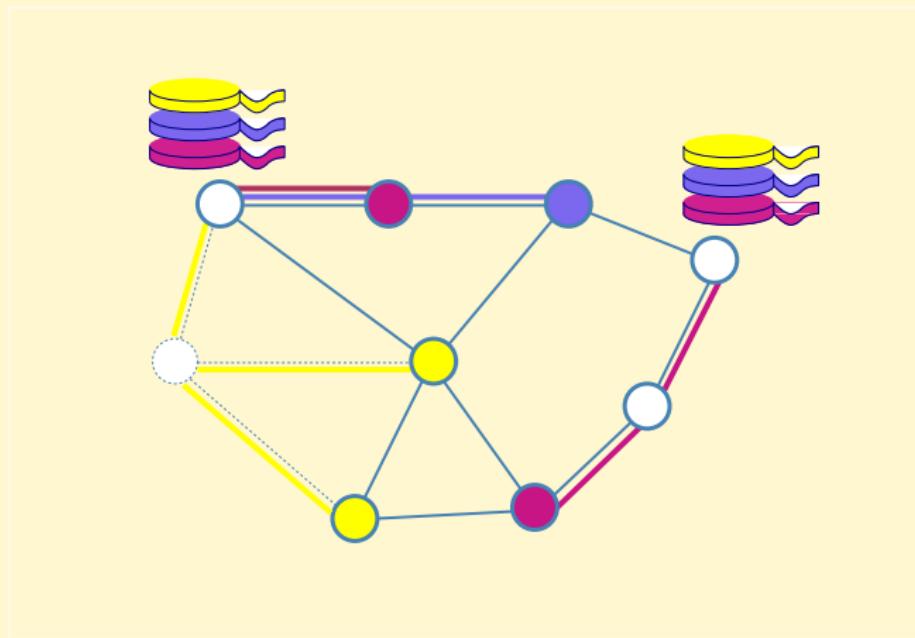
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- Self-stabilizing distributed algorithm



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

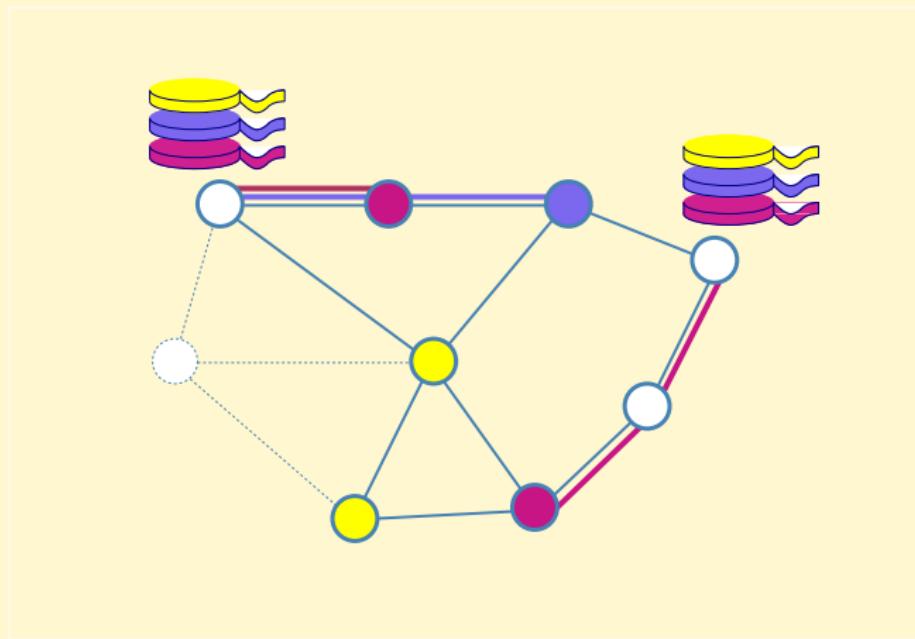
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- Self-stabilizing distributed algorithm



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

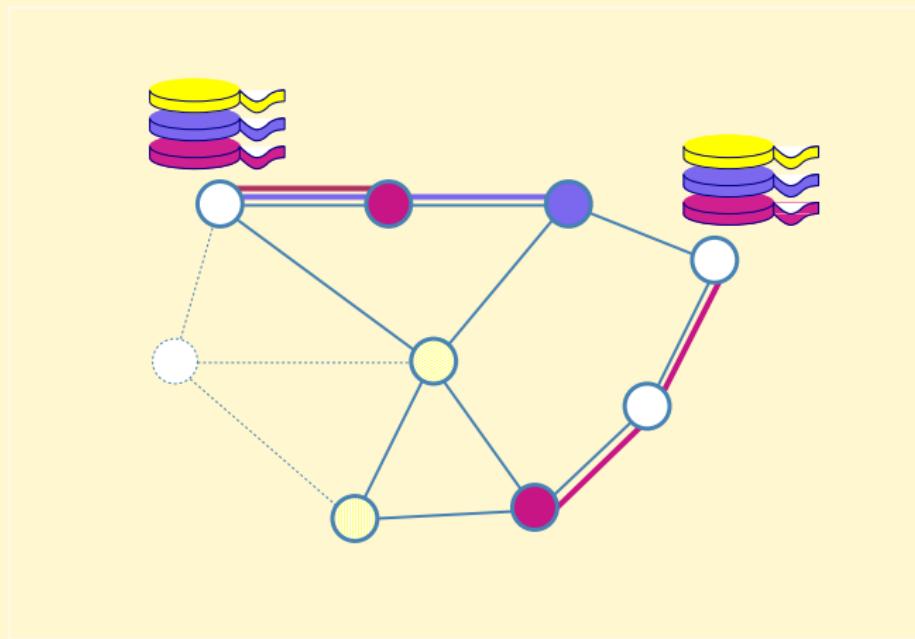
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- Self-stabilizing distributed algorithm



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

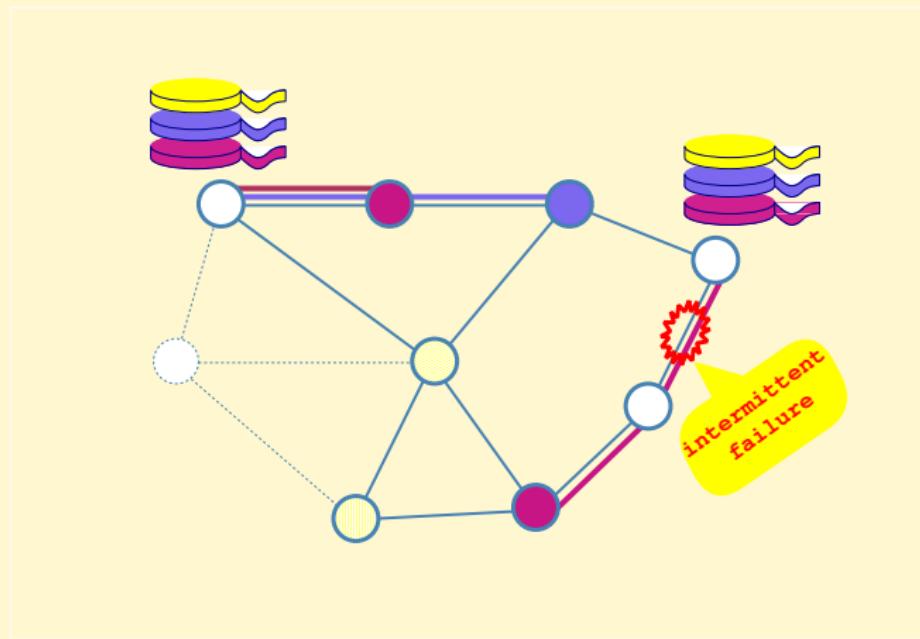
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- Self-stabilizing distributed algorithm



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

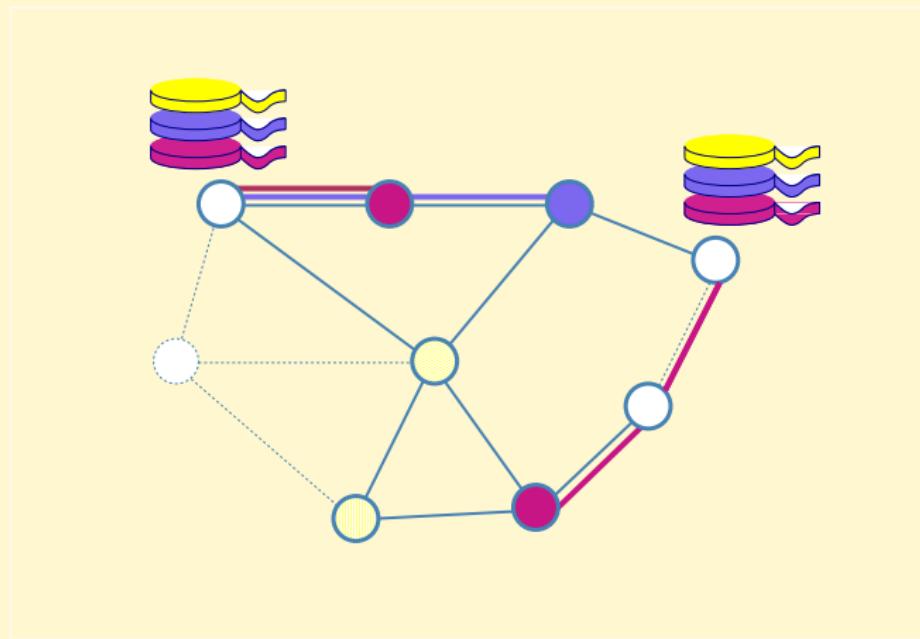
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- Self-stabilizing distributed algorithm



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks  
  
B. Ducourthial

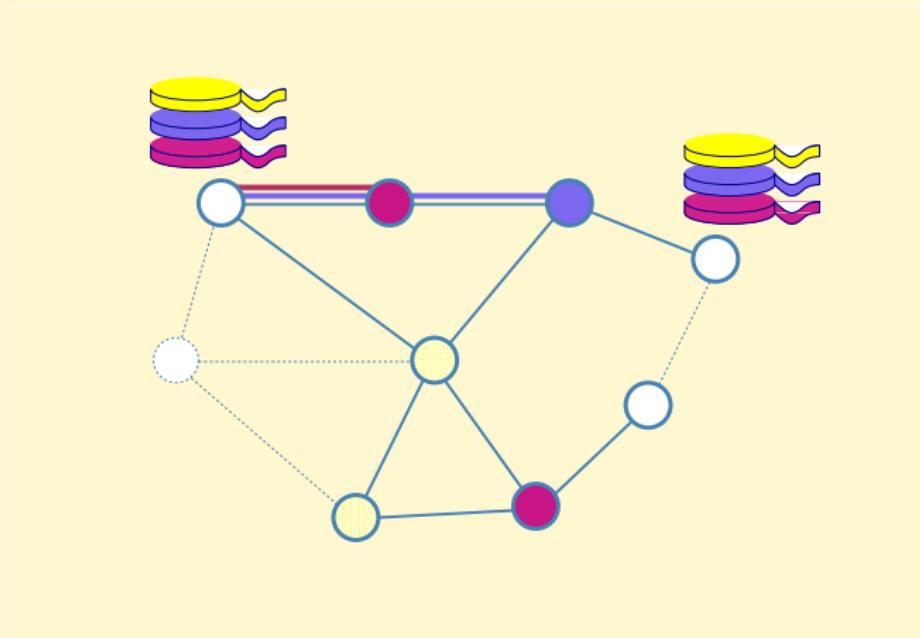
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- Self-stabilizing distributed algorithm



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

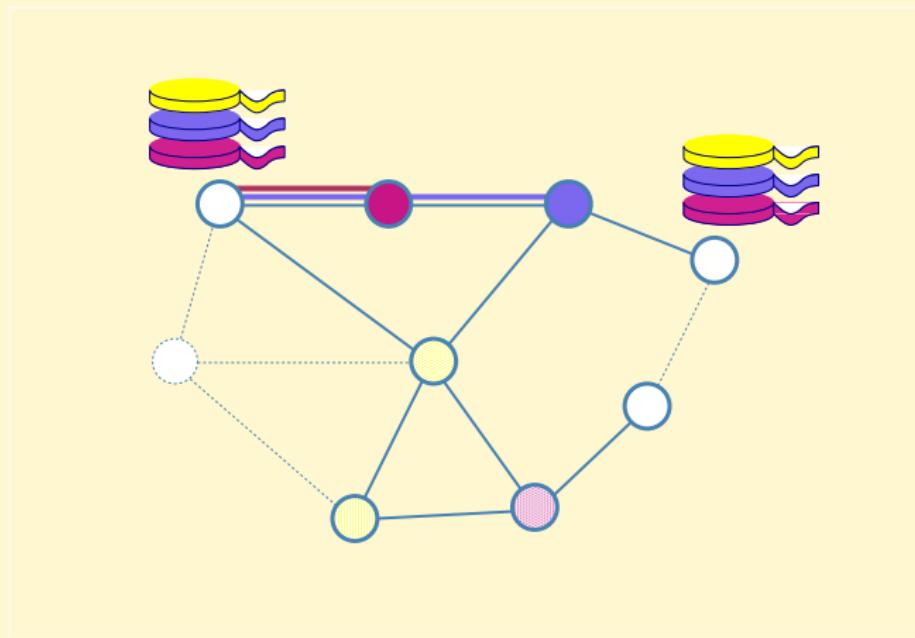
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- **Self-stabilizing distributed algorithm**



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

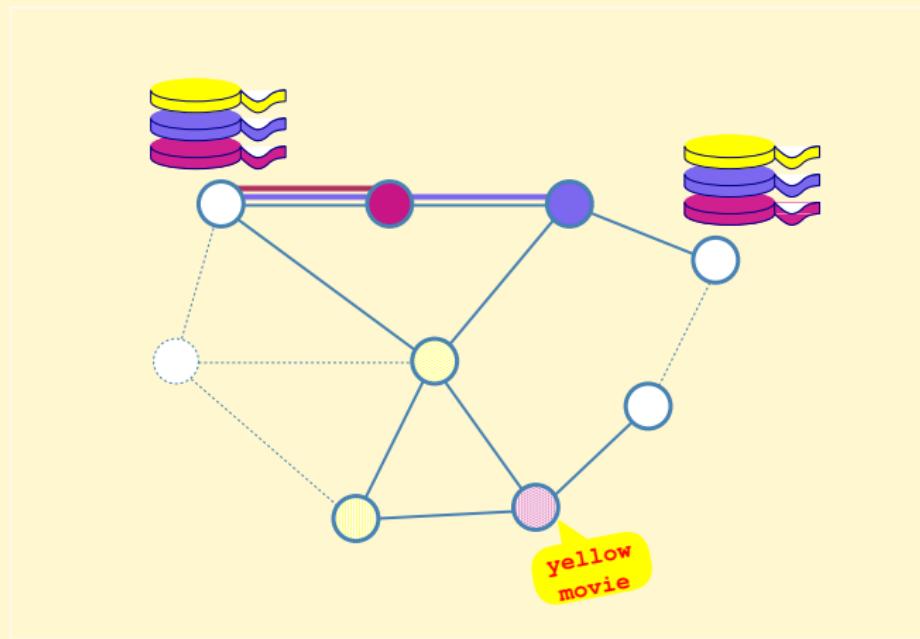
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- Self-stabilizing distributed algorithm



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

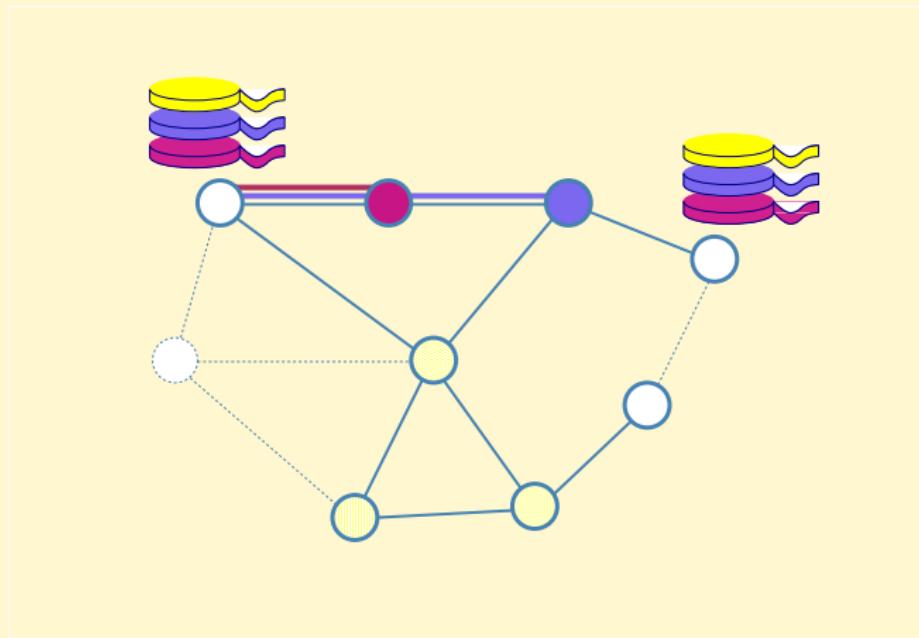
r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications

Distance  
Best paths  
Depth first  
Ancestors  
Routing

Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- Self-stabilizing distributed algorithm



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

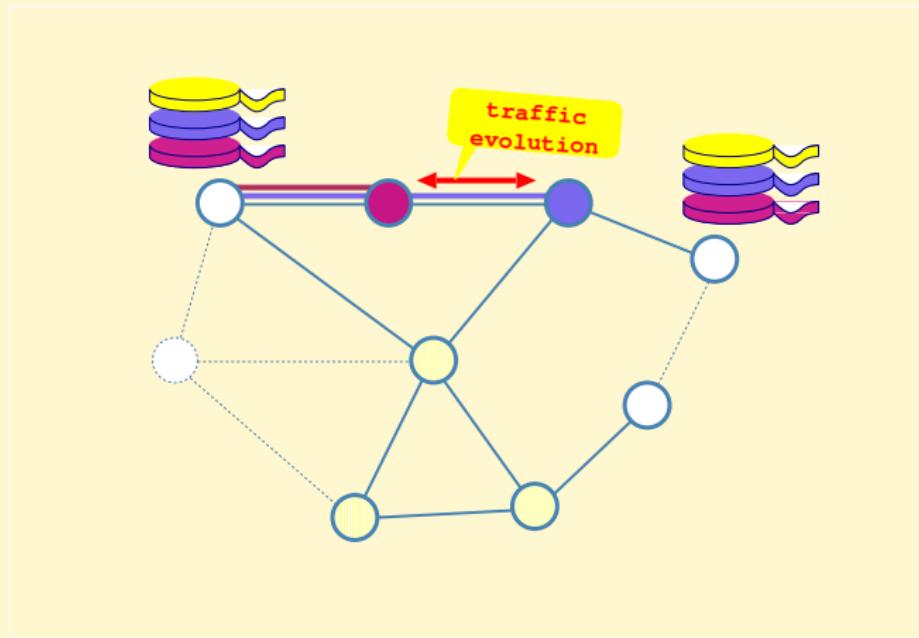
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- Self-stabilizing distributed algorithm



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

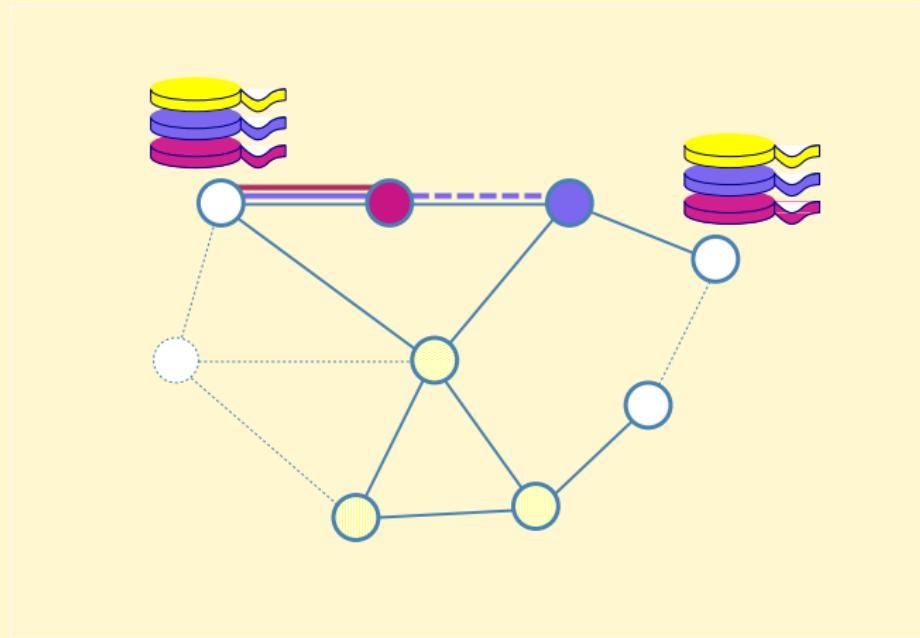
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- Self-stabilizing distributed algorithm



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks  
B. Ducourthial

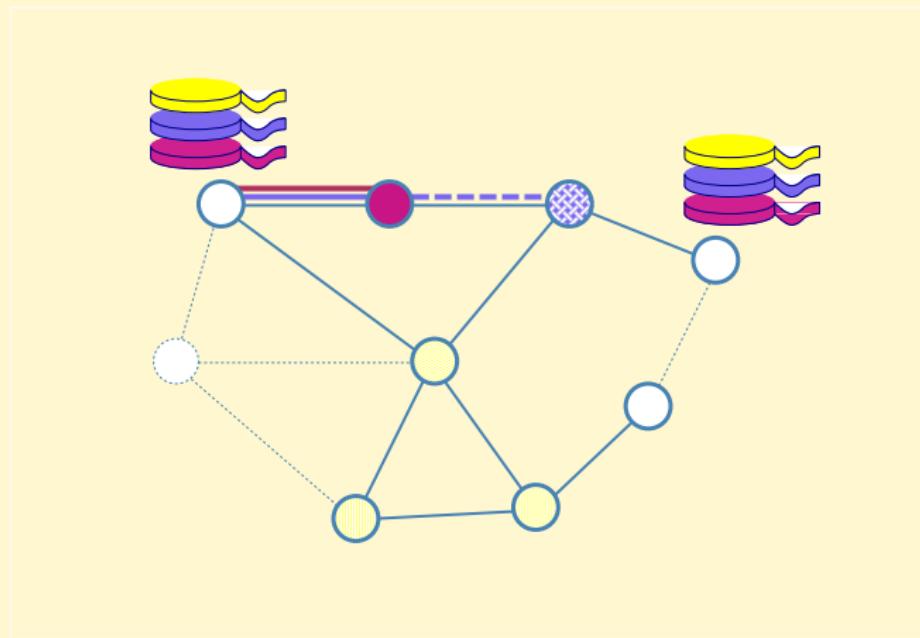
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- Self-stabilizing distributed algorithm



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

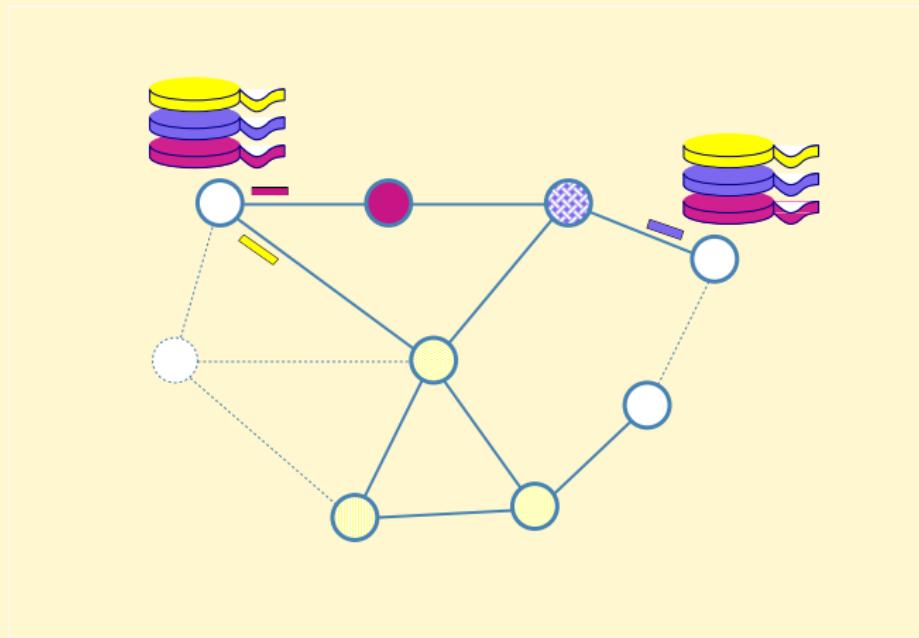
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- **Self-stabilizing distributed algorithm**



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

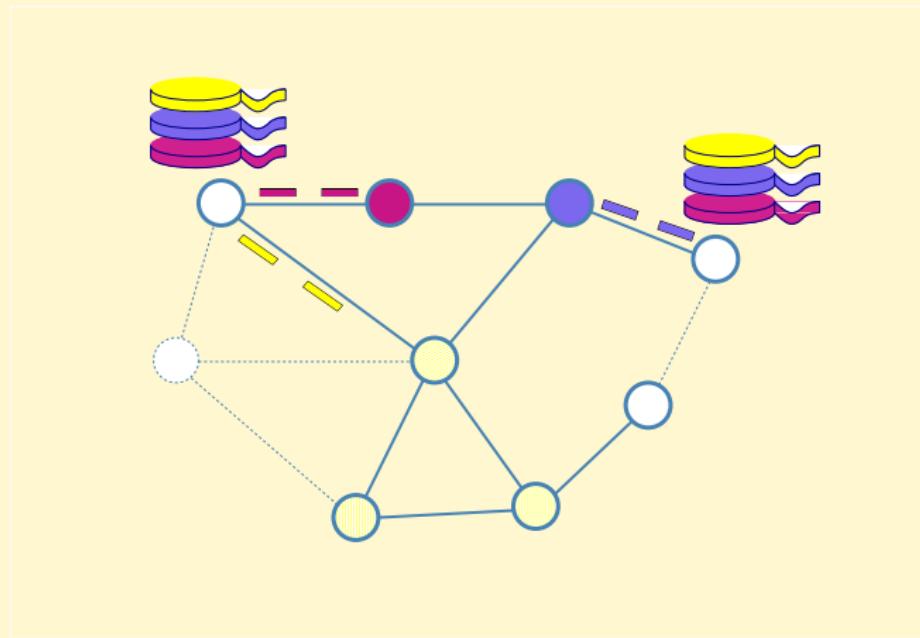
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- Self-stabilizing distributed algorithm



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

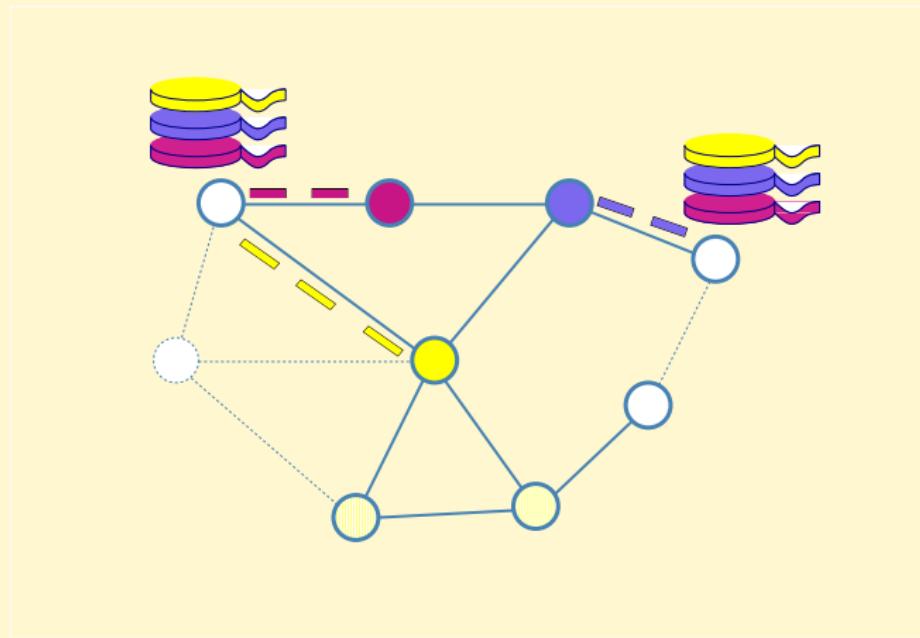
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- Self-stabilizing distributed algorithm



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network  
Computation  
Dyn. network  
Problem

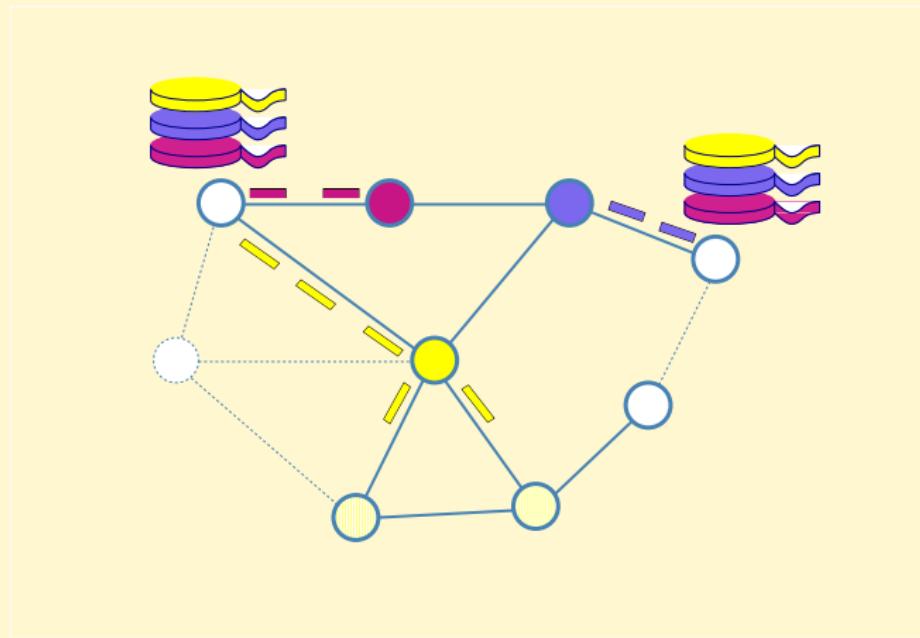
Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing

Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- Self-stabilizing distributed algorithm



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

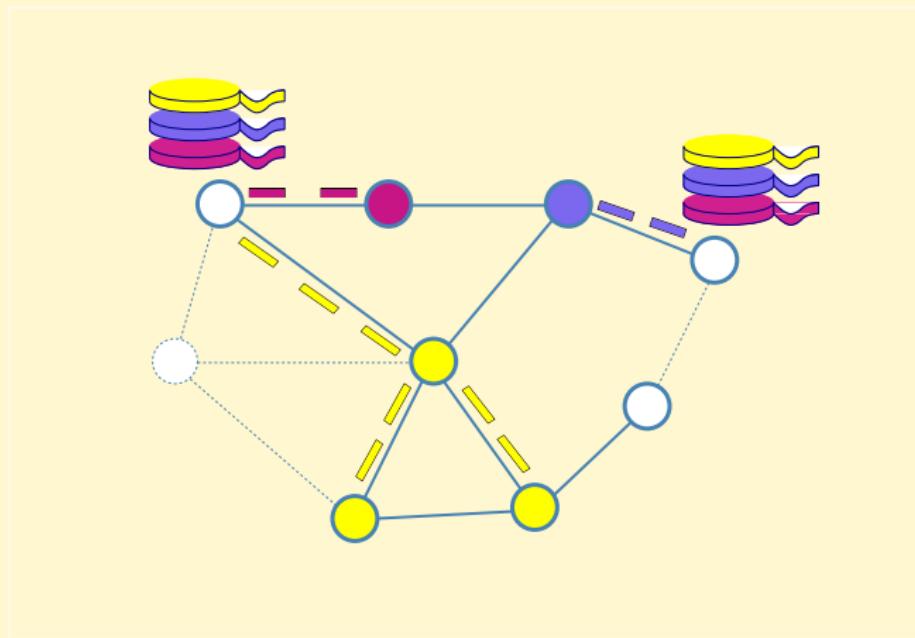
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths  $\rightsquigarrow$  strictly idempotent  $r$ -operator
- Groups of movies  $\rightsquigarrow$  union on acyclic graph
- Self-stabilizing distributed algorithm



# Applications of r-semi-groups

## Adaptive multisource multidistribution

Id. Algebra &  
Dyn. Networks

B. Ducourthial

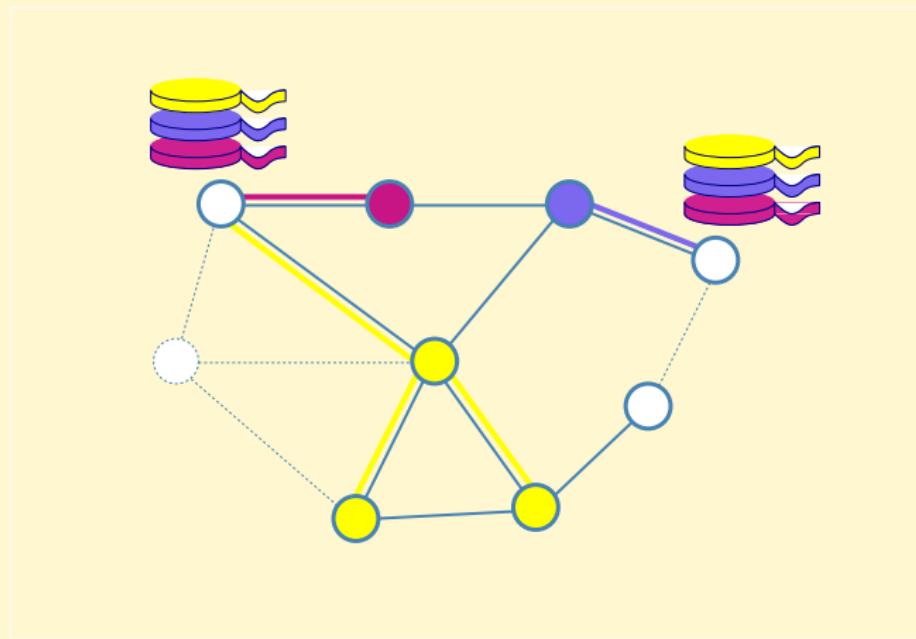
Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Best paths ↗ strictly idempotent  $r$ -operator
- Groups of movies ↗ union on acyclic graph
- **Self-stabilizing distributed algorithm**



# Applications of r-semi-groups

## Dynamic group in vehicular networks

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation

Dyn. network

Problem

Id. Algebra

Algorithm

Operator

Reminder

r-semi-group

Failure

r-semi-groups

Prerequisites

Definition

Properties

Applications

Distance

Best paths

Depth first

Ancestors

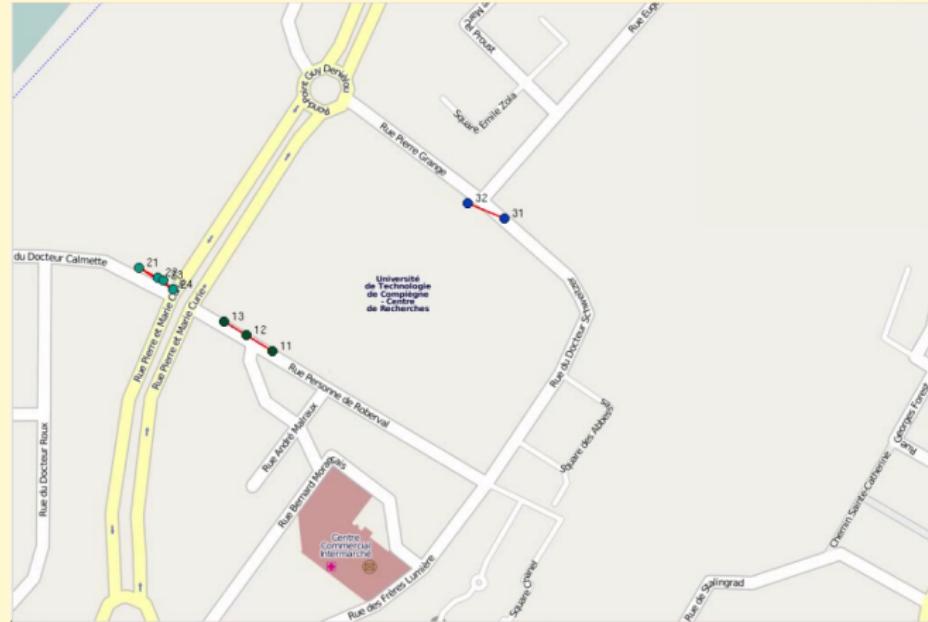
Routing

Broadcasting

Dyn. Group

Data collect

Conclusion



Road experiment replay using the Airplug Software Distribution

Clic on the image for loading the video (in the web browser)



# Applications of r-semi-groups

## Data collect in vehicular networks

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network  
Computation  
Dyn. network  
Problem

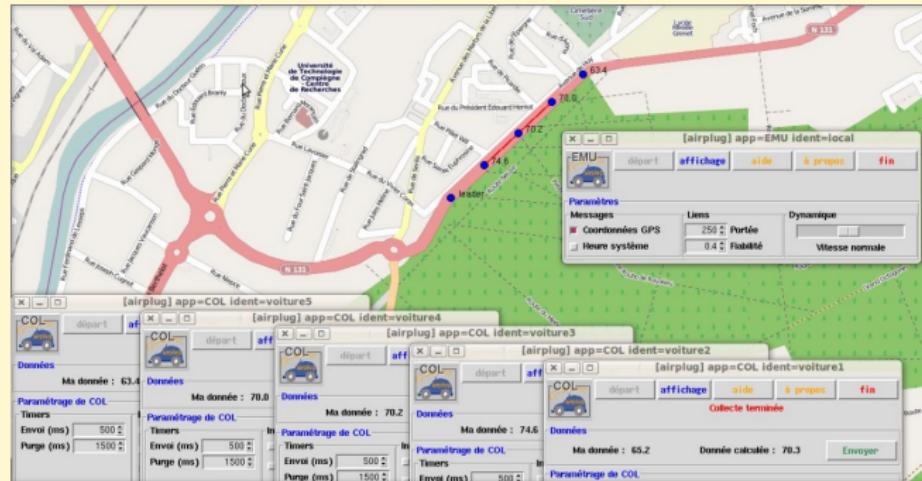
Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications

Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



Road experiment replay using the Airplug Software Distribution  
Clic on the image for loading the video (in the web browser)



# Summary

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network

Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



## 1 Computation in Dynamic Network



## 2 Using Idempotent Algebra



## 3 r-semi-group



## 4 Applications of r-semi-groups



## 5 Conclusion



# Conclusion

Id. Algebra &  
Dyn. Networks

B. Ducourthial

Dyn. Network  
Computation  
Dyn. network  
Problem

Id. Algebra  
Algorithm  
Operator  
Reminder

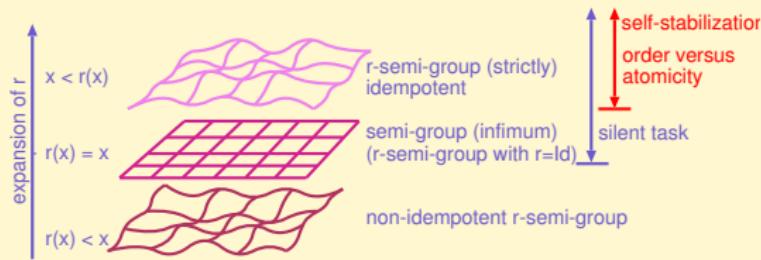
r-semi-group  
Failure  
r-semi-groups  
Prerequisites  
Definition  
Properties

Applications  
Distance  
Best paths  
Depth first  
Ancestors  
Routing  
Broadcasting  
Dyn. Group  
Data collect

Conclusion



- Designing stabilizing algorithms is not easy
- Particularly in case of failure/dynamic topology
- **r-semi-group:**
  - a generalization of the Abelian idempotent semi-group
  - generic proofs for a large family of applications



- More details and references at  
<https://www.hds.utc.fr/~ducourth>
- For movies, see: <https://www.hds.utc.fr/airplug>

