Modelling network performance with a spatial stochastic process algebra

Vashti Galpin
Laboratory for Foundations of Computer Science
University of Edinburgh

17 June 2010

Introduction

- model network performance

Introduction

- model network performance
- introduce spatial concepts to a stochastic process algebra

Introduction

- model network performance
- introduce spatial concepts to a stochastic process algebra
- analysis using continuous time Markov chains (CTMCs)

Introduction

- model network performance
- introduce spatial concepts to a stochastic process algebra
- analysis using continuous time Markov chains (CTMCs)
- demonstrate through an example

Introduction

- model network performance
- introduce spatial concepts to a stochastic process algebra
- analysis using continuous time Markov chains (CTMCs)
- demonstrate through an example
- other approaches to network modelling

Introduction

- model network performance
- introduce spatial concepts to a stochastic process algebra
- analysis using continuous time Markov chains (CTMCs)
- demonstrate through an example
- other approaches to network modelling
  - using the same spatial stochastic process algebra

Introduction

- model network performance
- introduce spatial concepts to a stochastic process algebra
- analysis using continuous time Markov chains (CTMCs)
- demonstrate through an example
- other approaches to network modelling
  - using the same spatial stochastic process algebra
  - using a process algebra with stochastic, continuous and discrete aspects

Introduction

- model network performance
- introduce spatial concepts to a stochastic process algebra
- analysis using continuous time Markov chains (CTMCs)
- demonstrate through an example
- other approaches to network modelling
  - using the same spatial stochastic process algebra
  - using a process algebra with stochastic, continuous and discrete aspects
- conclusions and further work

Motivation

- PEPA [Hillston 1996]

Motivation

- PEPA [Hillston 1996]
  - compact syntax, rules of behaviour

$$\frac{P \xrightarrow{(\alpha,r)} P'}{P + Q \xrightarrow{(\alpha,r)} P'}$$

Motivation

- PEPA [Hillston 1996]
  - compact syntax, rules of behaviour

$$\frac{P \xrightarrow{(\alpha,r)} P'}{P + Q \xrightarrow{(\alpha,r)} P'}$$

  - transitions labelled with $(\alpha, r) \in \mathcal{A} \times \mathbb{R}^+$

Motivation

- PEPA [Hillston 1996]
  - compact syntax, rules of behaviour

$$\frac{P \xrightarrow{(\alpha,r)} P'}{P + Q \xrightarrow{(\alpha,r)} P'}$$

  - transitions labelled with $(\alpha, r) \in \mathcal{A} \times \mathbb{R}^+$
  - interpret as continuous time Markov chain or ODEs

Motivation

- PEPA [Hillston 1996]
  - compact syntax, rules of behaviour

$$\frac{P \xrightarrow{(\alpha,r)} P'}{P + Q \xrightarrow{(\alpha,r)} P'}$$

  - transitions labelled with $(\alpha, r) \in \mathcal{A} \times \mathbb{R}^+$
  - interpret as continuous time Markov chain or ODEs
  - various analyses to understand performance

Motivation

- PEPA [Hillston 1996]
  - compact syntax, rules of behaviour

$$\frac{P \xrightarrow{(\alpha,r)} P'}{P + Q \xrightarrow{(\alpha,r)} P'}$$

  - transitions labelled with $(\alpha, r) \in \mathcal{A} \times \mathbb{R}^+$
  - interpret as continuous time Markov chain or ODEs
  - various analyses to understand performance
- add a general notion of location

Motivation

- PEPA [Hillston 1996]
  - compact syntax, rules of behaviour

$$\frac{P \xrightarrow{(\alpha,r)} P'}{P + Q \xrightarrow{(\alpha,r)} P'}$$

  - transitions labelled with $(\alpha, r) \in \mathcal{A} \times \mathbb{R}^+$
  - interpret as continuous time Markov chain or ODEs
  - various analyses to understand performance
- add a general notion of location
  - location names, cities

Motivation

- PEPA [Hillston 1996]
  - compact syntax, rules of behaviour

$$\frac{P \xrightarrow{(\alpha,r)} P'}{P + Q \xrightarrow{(\alpha,r)} P'}$$

  - transitions labelled with $(\alpha, r) \in \mathcal{A} \times \mathbb{R}^+$
  - interpret as continuous time Markov chain or ODEs
  - various analyses to understand performance
- add a general notion of location
  - location names, cities
  - points in n-dimensional space

Spatial stochastic process algebra

- locations, $\mathcal{L}$ and collections of locations, $\mathcal{P}_{\mathcal{L}}$

Spatial stochastic process algebra

- locations, $\mathcal{L}$ and collections of locations, $\mathcal{P}_{\mathcal{L}}$
- structure over $\mathcal{P}_{\mathcal{L}}$, weighted graph $G = (\mathcal{L}, E, w)$

Spatial stochastic process algebra

- locations, $\mathcal{L}$ and collections of locations, $\mathcal{P}_{\mathcal{L}}$
- structure over $\mathcal{P}_{\mathcal{L}}$, weighted graph $G = (\mathcal{L}, E, w)$
  - undirected hypergraph or directed graph

Spatial stochastic process algebra

- locations, $\mathcal{L}$ and collections of locations, $\mathcal{P}_{\mathcal{L}}$
- structure over $\mathcal{P}_{\mathcal{L}}$, weighted graph $G = (\mathcal{L}, E, w)$
  - undirected hypergraph or directed graph
  - $E \subseteq \mathcal{P}_{\mathcal{L}}$ and $w : E \rightarrow \mathbb{R}$

Spatial stochastic process algebra

- locations, $\mathcal{L}$ and collections of locations, $\mathcal{P}_{\mathcal{L}}$
- structure over $\mathcal{P}_{\mathcal{L}}$, weighted graph $G = (\mathcal{L}, E, w)$
  - undirected hypergraph or directed graph
  - $E \subseteq \mathcal{P}_{\mathcal{L}}$ and $w : E \to \mathbb{R}$
  - weights modify rates on actions between locations

Spatial stochastic process algebra

- locations, $\mathcal{L}$ and collections of locations, $\mathcal{P}_{\mathcal{L}}$
- structure over $\mathcal{P}_{\mathcal{L}}$, weighted graph $G = (\mathcal{L}, E, w)$
  - undirected hypergraph or directed graph
  - $E \subseteq \mathcal{P}_{\mathcal{L}}$ and $w : E \to \mathbb{R}$
  - weights modify rates on actions between locations
- $L \in \mathcal{P}_{\mathcal{L}} \quad \alpha \in \mathcal{A} \quad M \subseteq \mathcal{A} \quad r > 0$

Spatial stochastic process algebra

- locations, $\mathcal{L}$ and collections of locations, $\mathcal{P}_\mathcal{L}$
- structure over $\mathcal{P}_\mathcal{L}$, weighted graph $G = (\mathcal{L}, E, w)$
  - undirected hypergraph or directed graph
  - $E \subseteq \mathcal{P}_\mathcal{L}$ and $w : E \rightarrow \mathbb{R}$
  - weights modify rates on actions between locations
- $L \in \mathcal{P}_\mathcal{L} \quad \alpha \in \mathcal{A} \quad M \subseteq \mathcal{A} \quad r > 0$
- sequential components

$$S ::= (\alpha @ L, r).S \mid S + S \mid C_s @ L$$

Spatial stochastic process algebra

- locations, $\mathcal{L}$ and collections of locations, $\mathcal{P}_\mathcal{L}$
- structure over $\mathcal{P}_\mathcal{L}$, weighted graph $G = (\mathcal{L}, E, w)$
  - undirected hypergraph or directed graph
  - $E \subseteq \mathcal{P}_\mathcal{L}$ and $w : E \rightarrow \mathbb{R}$
  - weights modify rates on actions between locations
- $L \in \mathcal{P}_\mathcal{L} \quad \alpha \in \mathcal{A} \quad M \subseteq \mathcal{A} \quad r > 0$
- sequential components

$$S ::= (\alpha@L, r).S \mid S + S \mid C_s@L$$

Spatial stochastic process algebra

- locations, $\mathcal{L}$ and collections of locations, $\mathcal{P}_{\mathcal{L}}$
- structure over $\mathcal{P}_{\mathcal{L}}$, weighted graph $G = (\mathcal{L}, E, w)$
  - undirected hypergraph or directed graph
  - $E \subseteq \mathcal{P}_{\mathcal{L}}$ and $w : E \rightarrow \mathbb{R}$
  - weights modify rates on actions between locations
- $L \in \mathcal{P}_{\mathcal{L}} \quad \alpha \in \mathcal{A} \quad M \subseteq \mathcal{A} \quad r > 0$
- sequential components

$$S ::= (\alpha@L, r).S \mid S + S \mid C_s@L$$

- locations defined at sequential level only

Spatial stochastic process algebra

- locations, $\mathcal{L}$ and collections of locations, $\mathcal{P}_{\mathcal{L}}$
- structure over $\mathcal{P}_{\mathcal{L}}$, weighted graph $G = (\mathcal{L}, E, w)$
  - undirected hypergraph or directed graph
  - $E \subseteq \mathcal{P}_{\mathcal{L}}$ and $w : E \rightarrow \mathbb{R}$
  - weights modify rates on actions between locations
- $L \in \mathcal{P}_{\mathcal{L}} \quad \alpha \in \mathcal{A} \quad M \subseteq \mathcal{A} \quad r > 0$
- sequential components

$$S ::= (\alpha @ L, r).S \mid S + S \mid C_s @ L$$

- locations defined at sequential level only
- model components

$$P ::= P \bowtie_M P \mid P/M \mid C$$

Parameterised operational semantics

- define abstract process algebra parameterised by three functions

Parameterised operational semantics

- define abstract process algebra parameterised by three functions
- transitions labelled with $\mathcal{A} \times \mathcal{P}_{\mathcal{L}} \times \mathbb{R}^{+}$

Parameterised operational semantics

- define abstract process algebra parameterised by three functions
- transitions labelled with $\mathcal{A} \times \mathcal{P}_{\mathcal{L}} \times \mathbb{R}^+$
- Prefix

$$\frac{}{(\alpha @ L, r).S \xrightarrow{(\alpha @ L', r)} S} \quad L' = \mathit{apref}((\alpha @ L, r).S)$$

Parameterised operational semantics

- define abstract process algebra parameterised by three functions
- transitions labelled with $\mathcal{A} \times \mathcal{P}_{\mathcal{L}} \times \mathbb{R}^+$
- Prefix

$$\frac{}{(\alpha @ L, r).S \xrightarrow{(\alpha @ L', r)} S} \quad L' = \mathit{apref}((\alpha @ L, r).S)$$

- Cooperation $$\frac{P_1 \xrightarrow{(\alpha @ L_1, r_1)} P_1' \quad P_2 \xrightarrow{(\alpha @ L_2, r_2)} P_2'}{P_1 \underset{M}{\bowtie} P_2 \xrightarrow{(\alpha @ L, R)} P_1' \underset{M}{\bowtie} P_2'} \quad \alpha \in M$$

$$L = \mathit{async}(P_1, P_2, L_1, L_2) \quad R = \mathit{rsync}(P_1, P_2, L_1, L_2, r_1, r_2)$$

Parameterised operational semantics

- define abstract process algebra parameterised by three functions
- transitions labelled with $\mathcal{A} \times \mathcal{P}_{\mathcal{L}} \times \mathbb{R}^+$
- Prefix

$$\frac{}{(\alpha@L, r).S \xrightarrow{(\alpha@L', r)} S} \quad L' = \mathit{apref}((\alpha@L, r).S)$$

- Cooperation $$\frac{P_1 \xrightarrow{(\alpha@L_1, r_1)} P_1' \quad P_2 \xrightarrow{(\alpha@L_2, r_2)} P_2'}{P_1 \underset{M}{\bowtie} P_2 \xrightarrow{(\alpha@L, R)} P_1' \underset{M}{\bowtie} P_2'} \quad \alpha \in M$$

$$L = \mathit{async}(P_1, P_2, L_1, L_2) \quad R = \mathit{rsync}(P_1, P_2, L_1, L_2, r_1, r_2)$$

- other rules defined in the obvious manner

Parameterised operational semantics

- define abstract process algebra parameterised by three functions
- transitions labelled with $\mathcal{A} \times \mathcal{P}_{\mathcal{L}} \times \mathbb{R}^+$
- Prefix

$$\frac{}{(\alpha@L, r).S \xrightarrow{(\alpha@L', r)} S} \quad L' = \mathit{apref}((\alpha@L, r).S)$$

- Cooperation $\dfrac{P_1 \xrightarrow{(\alpha@L_1, r_1)} P_1' \quad P_2 \xrightarrow{(\alpha@L_2, r_2)} P_2'}{P_1 \underset{M}{\bowtie} P_2 \xrightarrow{(\alpha@L, R)} P_1' \underset{M}{\bowtie} P_2'} \quad \alpha \in M$

$$L = \mathit{async}(P_1, P_2, L_1, L_2) \quad R = \mathit{rsync}(P_1, P_2, L_1, L_2, r_1, r_2)$$

- other rules defined in the obvious manner
- instantiate functions to obtain concrete process algebra

Concrete process algebra for modelling networks

- networking performance

Concrete process algebra for modelling networks

- networking performance
- scenario

Concrete process algebra for modelling networks

- networking performance
- scenario
  - arbitrary topology

Concrete process algebra for modelling networks

- networking performance
- scenario
  - arbitrary topology
  - single packet traversal through network

Concrete process algebra for modelling networks

- networking performance
- scenario
  - arbitrary topology
  - single packet traversal through network
  - processes can be colocated

Concrete process algebra for modelling networks

- networking performance
- scenario
  - arbitrary topology
  - single packet traversal through network
  - processes can be colocated
- want to model different topologies and traffic

Concrete process algebra for modelling networks

- networking performance
- scenario
  - arbitrary topology
  - single packet traversal through network
  - processes can be colocated
- want to model different topologies and traffic
- choose functions to create process algebra

Concrete process algebra for modelling networks

- networking performance
- scenario
  - arbitrary topology
  - single packet traversal through network
  - processes can be colocated
- want to model different topologies and traffic
- choose functions to create process algebra
  - each sequential component must have single fixed location

Concrete process algebra for modelling networks

- networking performance
- scenario
  - arbitrary topology
  - single packet traversal through network
  - processes can be colocated
- want to model different topologies and traffic
- choose functions to create process algebra
  - each sequential component must have single fixed location
  - communication must be pairwise and directional

Concrete process algebra for modelling networks

- networking performance
- scenario
  - arbitrary topology
  - single packet traversal through network
  - processes can be colocated
- want to model different topologies and traffic
- choose functions to create process algebra
  - each sequential component must have single fixed location
  - communication must be pairwise and directional
- let $\mathcal{P}_{\mathcal{L}} = \mathcal{L} \cup (\mathcal{L} \times \mathcal{L})$, singletons and ordered pairs

Functions for concrete process algebra

- functions

Functions for concrete process algebra

- functions

$$apref(S) = \begin{cases} \ell & \text{if } ploc(S) = \{\ell\} \\ \bot & \text{otherwise} \end{cases}$$

Functions for concrete process algebra

- functions

$$apref(S) = \begin{cases} \ell & \text{if } ploc(S) = \{\ell\} \\ \bot & \text{otherwise} \end{cases}$$

$$async(P_1, P_2, L_1, L_2) = \begin{cases} (\ell_1, \ell_2) & \text{if } L_1 = \{\ell_1\}, L_2 = \{\ell_2\}, (\ell_1, \ell_2) \in E \\ \bot & \text{otherwise} \end{cases}$$

Functions for concrete process algebra

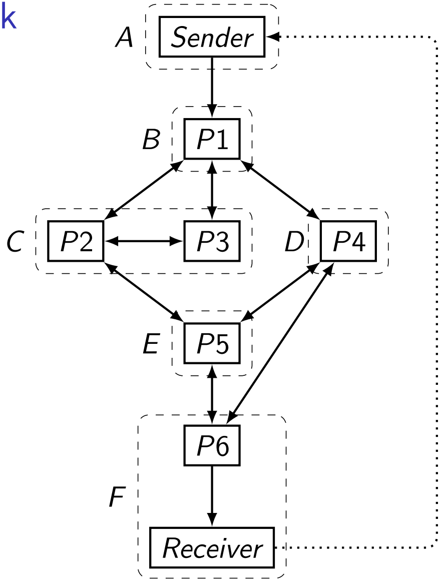
- functions

$$apref(S) = \begin{cases} \ell & \text{if } ploc(S) = \{\ell\} \\ \perp & \text{otherwise} \end{cases}$$

$$async(P_1, P_2, L_1, L_2) = \begin{cases} (\ell_1, \ell_2) & \text{if } L_1 = \{\ell_1\}, L_2 = \{\ell_2\}, (\ell_1, \ell_2) \in E \\ \perp & \text{otherwise} \end{cases}$$

$$rsync(P_1, P_2, L_1, L_2, r_1, r_2) = \begin{cases} \dfrac{r_1}{r_\alpha(P_1)} \dfrac{r_2}{r_\alpha(P_2)} \min(r_\alpha(P_1), r_\alpha(P_2)) \cdot w((\ell_1, \ell_2)) & \\ \qquad \text{if } L_1 = \{\ell_1\}, L_2 = \{\ell_2\}, (\ell_1, \ell_2) \in E \\ \perp \qquad \text{otherwise} \end{cases}$$

Example network



PEPA model

$$Sender@A \stackrel{def}{=} (prepare, \rho).Sending@A$$
$$Sending@A \stackrel{def}{=} \sum_{i=1}^{6}(c_{Si}, r_S).(ack, r_{ack}).Sender@A$$

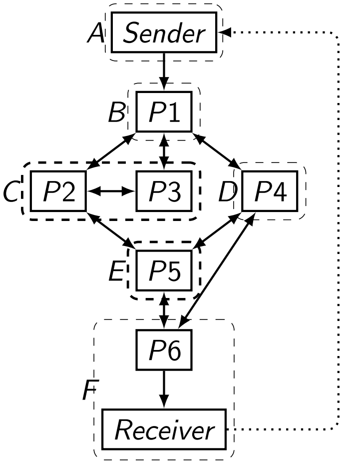
$$Receiver@F \stackrel{def}{=} \sum_{i=1}^{6}(c_{iR}, r_6).Receiving@F$$
$$Receiving@F \stackrel{def}{=} (consume, \gamma).(ack, r_{ack}).Receiver@F$$

$$P_i@\ell_i \stackrel{def}{=} (c_{Si}, \top).Q_i@\ell_i + \sum_{j=1,j\neq i}^{6}(c_{ji}, r).Q_i@\ell_i$$
$$Q_i@\ell_i \stackrel{def}{=} (c_{iR}, \top).P_i@\ell_i + \sum_{j=1,j\neq i}^{6}(c_{ij}, r).P_i@\ell_i$$

$$Network \stackrel{def}{=} (Sender@A \underset{*}{\bowtie} (P1@B \underset{*}{\bowtie} (P2@C \underset{*}{\bowtie} (P3@C \underset{*}{\bowtie} (P4@D \underset{*}{\bowtie} (P5@E \underset{*}{\bowtie} (P6@F \underset{*}{\bowtie} Receiver@F)))))))$$

Graphs

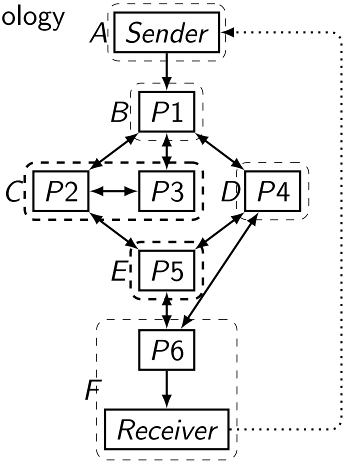
- rates: $r = r_R = r_S = 10$



Graphs

- rates: $r = r_R = r_S = 10$
- the weighted graph G describes the topology

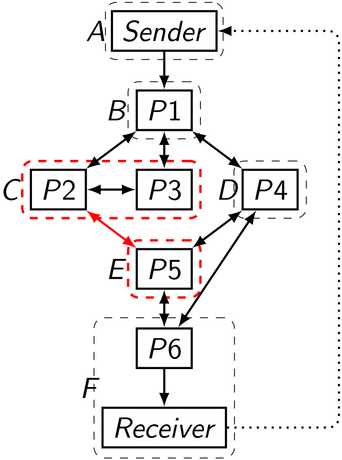
	A	B	C	D	E	F
A	1	1				
B			1	1		
C		1	1		1	
D		1			1	1
E			1	1		1
F	1			1	1	1



Graphs

- G_1 represents heavy traffic between C and E

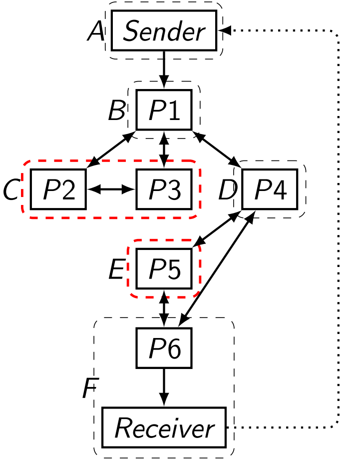
	A	B	C	D	E	F
A	1	1				
B			1	1		
C		1	1		0.1	
D		1			1	1
E			0.1	1		1
F	1			1	1	1



Graphs

- G_2 represents no connectivity between C and E

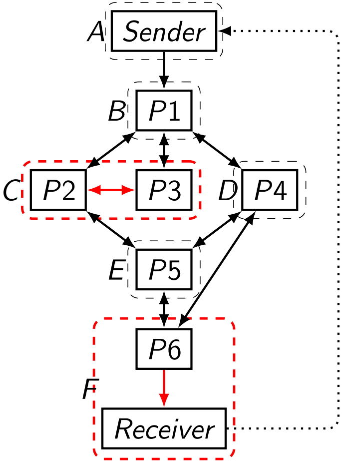
	A	B	C	D	E	F
A	1	1				
B			1	1		
C		1	1		0	
D		1			1	1
E			0	1		1
F	1			1	1	1



Graphs

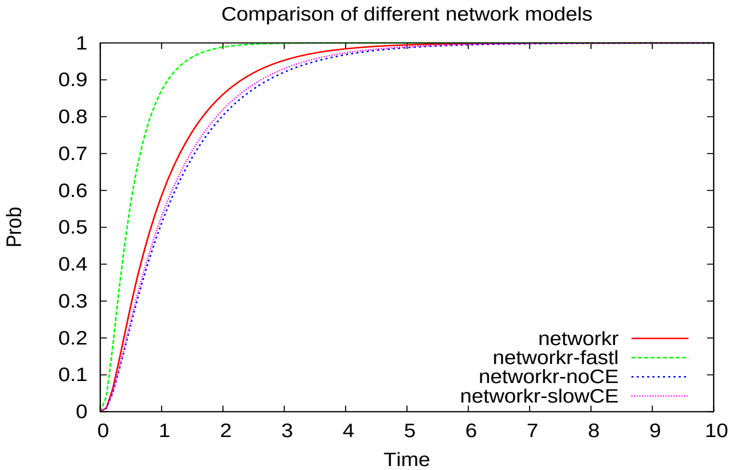
- G_3 represents high connectivity between colocated processes

	A	B	C	D	E	F
A	1	1				
B			1	1		
C		1	10		1	
D		1			1	1
E			1	1		1
F	1			1	1	10



Analysis

- cumulative density function of passage time



Evaluation

- uniform description for each node in the network

Evaluation

- uniform description for each node in the network
- network topology captured by graph

Evaluation

- uniform description for each node in the network
- network topology captured by graph
- graph modifications capture network variations

Evaluation

- uniform description for each node in the network
- network topology captured by graph
- graph modifications capture network variations
- existing analysis framework

Evaluation

- uniform description for each node in the network
- network topology captured by graph
- graph modifications capture network variations
- existing analysis framework
- abstract process algebra is flexible

Different concrete process algebras

- multiple packets

Different concrete process algebras

- multiple packets
  - each located node in network is one or more buffers

Different concrete process algebras

- multiple packets
  - each located node in network is one or more buffers
  - similar approach

Different concrete process algebras

- multiple packets
  - each located node in network is one or more buffers
  - similar approach
  - throughput, loss rates

Different concrete process algebras

- multiple packets
  - each located node in network is one or more buffers
  - similar approach
  - throughput, loss rates
- wireless sensor networks

Different concrete process algebras

- multiple packets
  - each located node in network is one or more buffers
  - similar approach
  - throughput, loss rates
- wireless sensor networks
  - actual physical location

Different concrete process algebras

- multiple packets
  - each located node in network is one or more buffers
  - similar approach
  - throughput, loss rates
- wireless sensor networks
  - actual physical location
  - weights capture performance characteristics over distance

Different concrete process algebras

- multiple packets
  - each located node in network is one or more buffers
  - similar approach
  - throughput, loss rates
- wireless sensor networks
  - actual physical location
  - weights capture performance characteristics over distance
- scope for many other scenarios

Different concrete process algebras

- multiple packets
  - each located node in network is one or more buffers
  - similar approach
  - throughput, loss rates
- wireless sensor networks
  - actual physical location
  - weights capture performance characteristics over distance
- scope for many other scenarios
  - different types of networks

Different concrete process algebras

- multiple packets
  - each located node in network is one or more buffers
  - similar approach
  - throughput, loss rates
- wireless sensor networks
  - actual physical location
  - weights capture performance characteristics over distance
- scope for many other scenarios
  - different types of networks
  - virus transmission in vineyards

And now for something slightly different

- Stochastic HYPE, joint with Jane Hillston and Luca Bortolussi

And now for something slightly different

- Stochastic HYPE, joint with Jane Hillston and Luca Bortolussi
- process algebra to model discrete, stochastic and continuous behaviour

And now for something slightly different

- Stochastic HYPE, joint with Jane Hillston and Luca Bortolussi
- process algebra to model discrete, stochastic and continuous behaviour
- semantic model
  - piecewise deterministic Markov processes
  - transition-driven stochastic hybrid automata

And now for something slightly different

- Stochastic HYPE, joint with Jane Hillston and Luca Bortolussi
- process algebra to model discrete, stochastic and continuous behaviour
- semantic model
  - piecewise deterministic Markov processes
  - transition-driven stochastic hybrid automata
- delay-tolerant networks

And now for something slightly different

- Stochastic HYPE, joint with Jane Hillston and Luca Bortolussi
- process algebra to model discrete, stochastic and continuous behaviour
- semantic model
  - piecewise deterministic Markov processes
  - transition-driven stochastic hybrid automata
- delay-tolerant networks
  - packets modelled as a continuous flow

And now for something slightly different

- Stochastic HYPE, joint with Jane Hillston and Luca Bortolussi
- process algebra to model discrete, stochastic and continuous behaviour
- semantic model
  - piecewise deterministic Markov processes
  - transition-driven stochastic hybrid automata
- delay-tolerant networks
  - packets modelled as a continuous flow
  - periods of connectivity modelled stochastically

And now for something slightly different

- Stochastic HYPE, joint with Jane Hillston and Luca Bortolussi
- process algebra to model discrete, stochastic and continuous behaviour
- semantic model
  - piecewise deterministic Markov processes
  - transition-driven stochastic hybrid automata
- delay-tolerant networks
  - packets modelled as a continuous flow
  - periods of connectivity modelled stochastically
  - full buffers modelled discretely

And now for something slightly different

- Stochastic HYPE, joint with Jane Hillston and Luca Bortolussi
- process algebra to model discrete, stochastic and continuous behaviour
- semantic model
  - piecewise deterministic Markov processes
  - transition-driven stochastic hybrid automata
- delay-tolerant networks
  - packets modelled as a continuous flow
  - periods of connectivity modelled stochastically
  - full buffers modelled discretely
  - determine storage required at nodes

Conclusion and further work

- conclusion

Conclusion and further work

- conclusion
  - stochastic process algebra with location

Conclusion and further work

- conclusion
  - stochastic process algebra with location
  - designed to be flexible

Conclusion and further work

- conclusion
  - stochastic process algebra with location
  - designed to be flexible
  - useful for modelling network performance

Conclusion and further work

- conclusion
  - stochastic process algebra with location
  - designed to be flexible
  - useful for modelling network performance
- further research

Conclusion and further work

- conclusion
  - stochastic process algebra with location
  - designed to be flexible
  - useful for modelling network performance
- further research
  - explore how it can be applied in modelling networks

Conclusion and further work

- conclusion
  - stochastic process algebra with location
  - designed to be flexible
  - useful for modelling network performance
- further research
  - explore how it can be applied in modelling networks
  - explore how it can be applied elsewhere

Conclusion and further work

- conclusion
  - stochastic process algebra with location
  - designed to be flexible
  - useful for modelling network performance
- further research
  - explore how it can be applied in modelling networks
  - explore how it can be applied elsewhere
  - comparison with other location-based formalism

Conclusion and further work

- conclusion
  - stochastic process algebra with location
  - designed to be flexible
  - useful for modelling network performance
- further research
  - explore how it can be applied in modelling networks
  - explore how it can be applied elsewhere
  - comparison with other location-based formalism
  - theoretical results for abstract process algebra

Conclusion and further work

- conclusion
  - stochastic process algebra with location
  - designed to be flexible
  - useful for modelling network performance
- further research
  - explore how it can be applied in modelling networks
  - explore how it can be applied elsewhere
  - comparison with other location-based formalism
  - theoretical results for abstract process algebra
  - behavioural equivalences

Thank you

This research was funded by the EPSRC SIGNAL Project

More comments

- related research

More comments

- related research
  - PEPA nets (Gilmore *et al*)

More comments

- related research
  - PEPA nets (Gilmore *et al*)
  - StoKlaim (de Nicola *et al*)

More comments

- related research
  - PEPA nets (Gilmore *et al*)
  - StoKlaim (de Nicola *et al*)
  - biological models – BioAmbients, attributed π-calculus

More comments

- related research
  - PEPA nets (Gilmore *et al*)
  - StoKlaim (de Nicola *et al*)
  - biological models – BioAmbients, attributed π-calculus
- locations and collections of location

More comments

- related research
  - PEPA nets (Gilmore *et al*)
  - StoKlaim (de Nicola *et al*)
  - biological models – BioAmbients, attributed π-calculus
- locations and collections of location
  - $\mathcal{P}_{\mathcal{L}} = 2^{\mathcal{L}}$, powerset

More comments

- related research
  - PEPA nets (Gilmore *et al*)
  - StoKlaim (de Nicola *et al*)
  - biological models – BioAmbients, attributed π-calculus
- locations and collections of location
  - $\mathcal{P}_\mathcal{L} = 2^\mathcal{L}$, powerset
  - $\mathcal{P}_\mathcal{L} = \mathcal{L} \cup (\mathcal{L} \times \mathcal{L})$, singletons and ordered pairs

More comments

- related research
  - PEPA nets (Gilmore *et al*)
  - StoKlaim (de Nicola *et al*)
  - biological models – BioAmbients, attributed π-calculus
- locations and collections of location
  - $\mathcal{P}_{\mathcal{L}} = 2^{\mathcal{L}}$, powerset
  - $\mathcal{P}_{\mathcal{L}} = \mathcal{L} \cup (\mathcal{L} \times \mathcal{L})$, singletons and ordered pairs
- different choices for $\mathcal{P}_{\mathcal{L}}$ give different semantics

More comments

- related research
  - PEPA nets (Gilmore *et al*)
  - StoKlaim (de Nicola *et al*)
  - biological models – BioAmbients, attributed π-calculus
- locations and collections of location
  - $\mathcal{P}_{\mathcal{L}} = 2^{\mathcal{L}}$, powerset
  - $\mathcal{P}_{\mathcal{L}} = \mathcal{L} \cup (\mathcal{L} \times \mathcal{L})$, singletons and ordered pairs
- different choices for $\mathcal{P}_{\mathcal{L}}$ give different semantics
  - locations associated with processes and/or actions

More comments

- related research
  - PEPA nets (Gilmore *et al*)
  - StoKlaim (de Nicola *et al*)
  - biological models – BioAmbients, attributed π-calculus
- locations and collections of location
  - $\mathcal{P}_\mathcal{L} = 2^\mathcal{L}$, powerset
  - $\mathcal{P}_\mathcal{L} = \mathcal{L} \cup (\mathcal{L} \times \mathcal{L})$, singletons and ordered pairs
- different choices for $\mathcal{P}_\mathcal{L}$ give different semantics
  - locations associated with processes and/or actions
  - singleton locations versus multiple locations

More comments

- related research
  - PEPA nets (Gilmore *et al*)
  - StoKlaim (de Nicola *et al*)
  - biological models – BioAmbients, attributed π-calculus
- locations and collections of location
  - $\mathcal{P}_{\mathcal{L}} = 2^{\mathcal{L}}$, powerset
  - $\mathcal{P}_{\mathcal{L}} = \mathcal{L} \cup (\mathcal{L} \times \mathcal{L})$, singletons and ordered pairs
- different choices for $\mathcal{P}_{\mathcal{L}}$ give different semantics
  - locations associated with processes and/or actions
  - singleton locations versus multiple locations
- longer terms aims

More comments

- related research
  - PEPA nets (Gilmore *et al*)
  - StoKlaim (de Nicola *et al*)
  - biological models – BioAmbients, attributed π-calculus
- locations and collections of location
  - $\mathcal{P}_\mathcal{L} = 2^\mathcal{L}$, powerset
  - $\mathcal{P}_\mathcal{L} = \mathcal{L} \cup (\mathcal{L} \times \mathcal{L})$, singletons and ordered pairs
- different choices for $\mathcal{P}_\mathcal{L}$ give different semantics
  - locations associated with processes and/or actions
  - singleton locations versus multiple locations
- longer terms aims
  - prove results for parametric process algebra

More comments

- related research
  - PEPA nets (Gilmore *et al*)
  - StoKlaim (de Nicola *et al*)
  - biological models – BioAmbients, attributed π-calculus
- locations and collections of location
  - $\mathcal{P}_{\mathcal{L}} = 2^{\mathcal{L}}$, powerset
  - $\mathcal{P}_{\mathcal{L}} = \mathcal{L} \cup (\mathcal{L} \times \mathcal{L})$, singletons and ordered pairs
- different choices for $\mathcal{P}_{\mathcal{L}}$ give different semantics
  - locations associated with processes and/or actions
  - singleton locations versus multiple locations
- longer terms aims
  - prove results for parametric process algebra
  - then apply to concrete process algebra