nurbin

#### Introductio

κ syntax

#### Kappa at DLah

DLab's curren

Space-related

Timming contro

pre-Kappa

# pre-kappa expander for $\kappa$ language

Héctor Urbina

July 19, 2011

# Introduction What is κ

Kappa at OLab

DLab's current work Space-related simulations Timming contro pre-Kappa

- Introduction
  - ullet What is  $\kappa$
  - $\kappa$  syntax
- 2 Kappa at DLab
  - DLab's current work
  - Space-related simulations
  - Timming control
  - pre-Kappa expander

pre- $\kappa$  expander

hurbin

Introduction What is  $\kappa$ 

Nappa at DLab DLab's curre

DLab's currer work Space-related simulations

Timming conti pre-Kappa expander  $\kappa$  is a formal language for defining agents as sets of sites.

Sites hold an internal state as well as a binding state.

 $\kappa$  also enables the expression of rules of interaction between agents.

These rules are executable, inducing a stochastic dynamics on a mixture of agents.

A  $\kappa$  model is a collection of rules (with rate constants) and an initial mixture of agents on which such rules begin to act.

## pre- $\kappa$ expander

hurbin

Introductio

What is κ
κ syntax

Kappa at

DLab

DLab's curren
work

Space-related
simulations

Timming cont

 $\kappa$  is a formal language for defining agents as sets of sites.

Sites hold an internal state as well as a binding state.

 $\kappa$  also enables the expression of rules of interaction between agents.

These rules are executable, inducing a stochastic dynamics on a mixture of agents.

A  $\kappa$  model is a collection of rules (with rate constants) and an initial mixture of agents on which such rules begin to act.

## pre- $\kappa$ expander

hurbin

What is  $\kappa$   $\kappa$  syntax

Kappa at

DLab

DLab's currer
work

 $\kappa$  is a formal language for defining agents as sets of sites.

Sites hold an internal state as well as a binding state.

 $\kappa$  also enables the expression of rules of interaction between agents.

These rules are executable, inducing a stochastic dynamics on a mixture of agents.

A  $\kappa$  model is a collection of rules (with rate constants) and an initial mixture of agents on which such rules begin to act.

## pre- $\kappa$ expander

hurbin

What is  $\kappa$  syntax

Kappa at DLab

DLab's current work

Space-related simulations

Timming control rights and specific properties.

 $\kappa$  is a formal language for defining agents as sets of sites.

Sites hold an internal state as well as a binding state.

 $\kappa$  also enables the expression of rules of interaction between agents.

These rules are executable, inducing a stochastic dynamics on a mixture of agents.

A  $\kappa$  model is a collection of rules (with rate constants) and an initial mixture of agents on which such rules begin to act.

## pre- $\kappa$ expander

hurbin

What is  $\kappa$   $\kappa$  syntax

Kappa at
DLab

DLab's current
work

Space-related
simulations

Timming control

 $\kappa$  is a formal language for defining agents as sets of sites.

Sites hold an internal state as well as a binding state.

 $\kappa$  also enables the expression of rules of interaction between agents.

These rules are executable, inducing a stochastic dynamics on a mixture of agents.

A  $\kappa$  model is a collection of rules (with rate constants) and an initial mixture of agents on which such rules begin to act.

# $\kappa$ Syntax short introduction

pre- $\kappa$  expander

hurbin

Introductio
What is  $\kappa$   $\kappa$  syntax

DLab DLab's curre

DLab's curren work Space-related simulations Timming cont

simulations
Timming contro

### Rule in English:

"Unphosphorilated Site1 of A binds to Site1 of B."

 $\kappa$  Rule:

$$A(Site1^u), B(Site1) \rightarrow A(Site1^u!1), B(Site1!1)$$

- Agent Names : an identifier
- Agent Sites: an identifier
- Internal States : ~(value)
- Binding States : !(n), !\_ or ?.

# $\kappa$ Syntax short introduction

pre- $\kappa$  expander

hurbin

Introductio
What is  $\kappa$   $\kappa$  syntax

Kappa at DLab

DLab's current work Space-related simulations Timming cont Rule in English:

"Unphosphorilated Site1 of A binds to Site1 of B."

 $\kappa$  Rule:

 $\mathsf{A}(\mathsf{Site1}^\mathsf{-}\mathsf{u}), \mathsf{B}(\mathsf{Site1}) \to \mathsf{A}(\mathsf{Site1}^\mathsf{-}\mathsf{u}!1), \mathsf{B}(\mathsf{Site1}!1)$ 

- Agent Names: an identifier.
- Agent Sites: an identifier
- Internal States : ~(value).
- Binding States :  $!\langle n \rangle$ ,  $!_{-}$  or ?.

# $\kappa$ Syntax short introduction

## pre- $\kappa$ expander

hurbin

ntroductio
What is  $\kappa$   $\kappa$  syntax

# Kappa at

OLab's current work Space-related simulations Fimming contro ore-Kappa

### Rule in English:

"Unphosphorilated Site1 of A binds to Site1 of B."

### $\kappa$ Rule:

$$\mathsf{A}(\mathsf{Site1}^\mathsf{-}\mathsf{u}), \mathsf{B}(\mathsf{Site1}) \to \mathsf{A}(\mathsf{Site1}^\mathsf{-}\mathsf{u}!1), \mathsf{B}(\mathsf{Site1}!1)$$

- Agent Names : an identifier.
- Agent Sites: an identifier.
- Internal States : ~(value).
- Binding States :  $!\langle n \rangle$ ,  $!_-$  or ?.

# Kappa file structure

10000 C

%init:

 $pre-\kappa$ expander #### Signatures %agent: A(x,c) # Declaration of agent A %agent: B(x) # Declaration of B %agent: C(x1~u~p,x2~u~p) # Declaration of C with 2 modifiable sites 5 #### Rules 6 'a.b' A(x).B(x) -> A(x!1).B(x!1) @ 'on\_rate' #A binds B  $\alpha$  'a..b'  $A(x!1), B(x!1) \rightarrow A(x), B(x) @ 'off_rate' #AB dissociation'$  'ab.c' A(x!\_,c),C(x1~u) ->A(x!\_,c!2),C(x1~u!2) @ 'on\_rate' #AB binds C 'mod x1' C(x1~u!1).A(c!1) ->C(x1~p).A(c) @ 'mod\_rate' #AB modifies x1 'a.c' A(x,c),C(x1~p,x2~u) -> A(x,c!1),C(x1~p,x2~u!1) @ 'on\_rate' #A binds C on x2 'mod x2' A(x,c!1),C(x1~p,x2~u!1) -> A(x,c),C(x1~p,x2~p) @ 'mod\_rate' #A modifies x2 #### Variables %var: 'on\_rate' 1.0E-4 # per molecule per second %var: 'off\_rate' 0.1 # per second %var: 'mod\_rate' 1 # per second Mobs: 'AB' A(x!x.B) %obs: 'Cuu' C(x1~u.x2~u) %obs: 'Cpu' C(x1~p.x2~u) %obs: 'Cpp' C(x1~p.x2~p) #### Initial conditions %init: 1000 A,B

pre- $\kappa$  expander

hurbin

Introductio

What is  $\kappa$  $\kappa$  syntax

Kappa at DLab

DLab's current work

Space-related simulations

simulations Timming con

### DLab members study complex dynamical systems.

Currently, Cesar Ravello is modeling muscle contration and Felipe Nuñez is simulating massive responses to zombie attacks on human populations, whereas Ricardo Honorato is adapting Model Checking techniques to be used with systems expressed in  $\kappa$  language.

Without intervening the  $\kappa$  language, we have reached some interesting levels of abstraction!

 $pre-\kappa$ expander

DI ah's current

DLab members study complex dynamical systems.

Currently, Cesar Ravello is modeling muscle contration and Felipe Nuñez is simulating massive responses to zombie attacks on human populations, whereas Ricardo Honorato is adapting Model Checking techniques to be used with systems expressed in  $\kappa$  language.

pre- $\kappa$  expander

hurbin

Introduction What is  $\kappa$ 

Kappa a

DLab's current

Space-related simulations

simulations Timming con DLab members study complex dynamical systems.

Currently, Cesar Ravello is modeling muscle contration and Felipe Nuñez is simulating massive responses to zombie attacks on human populations, whereas Ricardo Honorato is adapting Model Checking techniques to be used with systems expressed in  $\kappa$  language.

Without intervening the  $\kappa$  language, we have reached some interesting levels of abstraction!

## pre- $\kappa$ expander

hurbin

#### Introduction

vvnat is κ κ syntax

Kappa a OLab

### DLab's current

work Space-related

simulations Timming contro

pre-Kappa expander

- Space-related simulations.
  - Compartmentalization.
    - Diffusion events.
- Timming control.
  - Polymer-driven rules to manipulate latency.

## pre- $\kappa$ expander

hurbin

#### Introduction

What is ε κ syntax

Kappa a OLab

### DLab's current

work Space-related

simulations Timming contro

pre-Kappa expander

- Space-related simulations.
  - Compartmentalization.
  - Diffusion events.
- Timming control.
  - Polymer-driven rules to manipulate latency.

## pre- $\kappa$ expander

hurbin

#### Introductio

κ syntax

Kappa at DLab

### DLab's current work

Space-related simulations Timming contro pre-Kappa • Space-related simulations.

- Compartmentalization.
- Diffusion events.
- Timming control.
  - Polymer-driven rules to manipulate latency.

pre- $\kappa$  expander

nurbin

#### Introductio

What is  $\kappa$ 

κ syntax

DLab's cur

DLab's currer work

### Space-related simulations

Timming contr

### **#Signatures**

%agent:  $A(x,c,loc^ij^k)$ %agent:  $B(x,loc^ij^k)$ 

#### #Rules

#### #A binds E

$$\begin{split} &\mathsf{A}(\mathsf{x},\mathsf{loc}\tilde{\ \ i}),\mathsf{B}(\mathsf{x},\mathsf{loc}\tilde{\ \ i}) \to \mathsf{A}(\mathsf{x}!1,\mathsf{loc}\tilde{\ \ i}),\mathsf{B}(\mathsf{x}!1,\mathsf{loc}\tilde{\ \ i}) \ @ \ \ \mathsf{'on\_rate'} \\ &\mathsf{A}(\mathsf{x},\mathsf{loc}\tilde{\ \ j}),\mathsf{B}(\mathsf{x},\mathsf{loc}\tilde{\ \ j}) \to \mathsf{A}(\mathsf{x}!1,\mathsf{loc}\tilde{\ \ j}),\mathsf{B}(\mathsf{x}!1,\mathsf{loc}\tilde{\ \ j}) \ @ \ \ \mathsf{'on\_rate'} \\ &\mathsf{A}(\mathsf{x},\mathsf{loc}\tilde{\ \ k}),\mathsf{B}(\mathsf{x},\mathsf{loc}\tilde{\ \ k}) \to \mathsf{A}(\mathsf{x}!1,\mathsf{loc}\tilde{\ \ k}),\mathsf{B}(\mathsf{x}!1,\mathsf{loc}\tilde{\ \ k}) \ @ \ \ \mathsf{'on\_rate'} \end{split}$$

pre- $\kappa$  expander

hurbin

#### Introductio

What is  $\kappa$   $\kappa$  syntax

Kappa at

DLab's curre

work
Space-related

simulations

Timming cont pre-Kappa

```
#Signatures
```

%agent:  $A(x,c,loc^ij^k)$ %agent:  $B(x,loc^ij^k)$ 

### #Rules

#### #A binds B

$$\begin{split} &\mathsf{A}(\mathsf{x},\mathsf{loc}\tilde{\ i}),\mathsf{B}(\mathsf{x},\mathsf{loc}\tilde{\ i}) \to \mathsf{A}(\mathsf{x}|1,\mathsf{loc}\tilde{\ i}),\mathsf{B}(\mathsf{x}|1,\mathsf{loc}\tilde{\ i}) \ @ \ \mathsf{'on\_rate'} \\ &\mathsf{A}(\mathsf{x},\mathsf{loc}\tilde{\ i}),\mathsf{B}(\mathsf{x},\mathsf{loc}\tilde{\ i}) \to \mathsf{A}(\mathsf{x}|1,\mathsf{loc}\tilde{\ i}),\mathsf{B}(\mathsf{x}|1,\mathsf{loc}\tilde{\ i}) \ @ \ \mathsf{'on\_rate'} \\ &\mathsf{A}(\mathsf{x},\mathsf{loc}\tilde{\ k}),\mathsf{B}(\mathsf{x},\mathsf{loc}\tilde{\ k}) \to \mathsf{A}(\mathsf{x}|1,\mathsf{loc}\tilde{\ k}),\mathsf{B}(\mathsf{x}|1,\mathsf{loc}\tilde{\ k}) \ @ \ \mathsf{'on\_rate'} \end{split}$$

pre- $\kappa$  expander

hurbin

Introduction

What is κ κ syntax

Kappa at DLab

work Space-related

simulations

Timming contr pre-Kappa

```
#Locations i, j and k have different volumen/area!
```

#Signatures

%agent: A(x,c,loc~i~j~k)
%agent: B(x loc~i~i~k)

### #Rules

#### #A binds B

 $A(x,loc^{\sim}i),B(x,loc^{\sim}i) \rightarrow A(x!1,loc^{\sim}i),B(x!1,loc^{\sim}i) @ 'on\_rate\_loc(i)'$ 

 $A(x,loc^{-}j),B(x,loc^{-}j) \rightarrow A(x!1,loc^{-}j),B(X!1,loc^{-}j) @ 'on_rate_loc(j)'$ 

 $A(x,loc^*k),B(x,loc^*k) \rightarrow A(x!1,loc^*k),B(X!1,loc^*k) @ 'on\_rate\_loc(k)$ 

#### #AB dissociation

 $A(x!1,loc^{\sim}i),B(x!1,loc^{\sim}i) \rightarrow A(x,loc^{\sim}i),B(x,loc^{\sim}i)$  @ 'off\_rate'

 $A(x!1,loc^{-}j),B(x!1,loc^{-}j) \rightarrow A(x,loc^{-}j),B(x,loc^{-}j)$  @ 'off\_rate'

 $A(x!1,loc^k),B(x!1,loc^k) \rightarrow A(x,loc^k),B(x,loc^k)$  @ 'off\_rate

```
pre-\kappa expander
```

hurbin

Introduction

κ syntax

Kappa at DLab

DLab's curre

Space-related simulations

Timming cont

```
#Locations i, j and k have different volumen/area! #Signatures %agent: A(x,c,loc^{-i})^{-i}k) %agent: B(x,loc^{-i})^{-i}k) #Rules #A binds B A(x,loc^{-i}),B(x,loc^{-i}) \rightarrow A(x!1,loc^{-i}),B(x!1,loc^{-i}) @ 'on_rate_loc(i)' A(x,loc^{-j}),B(x,loc^{-j}) \rightarrow A(x!1,loc^{-j}),B(X!1,loc^{-j}) @ 'on_rate_loc(j)' A(x,loc^{-k}),B(x,loc^{-k}) \rightarrow A(x!1,loc^{-k}),B(X!1,loc^{-k}) @ 'on_rate_loc(k)'
```

```
pre-\kappa expander
```

hurbin

Introduction What is  $\kappa$ 

Kappa at OLab

DLab's curre work

Space-related simulations

Timming cont

```
#Locations i, j and k have different volumen/area!
#Signatures
%agent: A(x,c,loc~i~i~k)
%agent: B(x,loc^i^j^k)
#Rules
#A binds B
A(x,loc^{-i}),B(x,loc^{-i}) \rightarrow A(x!1,loc^{-i}),B(x!1,loc^{-i}) @ 'on_rate_loc(i)'
A(x,loc^*i),B(x,loc^*i) \rightarrow A(x!1,loc^*i),B(X!1,loc^*i) @ 'on_rate_loc(i)'
A(x,loc^k),B(x,loc^k) \rightarrow A(x!1,loc^k),B(X!1,loc^k) @ 'on_rate_loc(k)'
#AB dissociation
A(x!1,loc^{-i}),B(x!1,loc^{-i}) \rightarrow A(x,loc^{-i}),B(x,loc^{-i}) @ 'off_rate'
A(x!1,loc^*i),B(x!1,loc^*j) \rightarrow A(x,loc^*j),B(x,loc^*j) @ 'off_rate'
```

 $A(x!1,loc^k),B(x!1,loc^k) \rightarrow A(x,loc^k),B(x,loc^k)$  @ 'off\_rate'

### Diffusion events

pre- $\kappa$  expander

nurbin

#### Introduction

What is κ κ syntax

#### Kappa at DLab

DLab's currer

### work Space-related

simulations Timming control

pre-Kappa

### **#Signatures**

%agent:  $A(x,c,loc^ij^k)$ 

%agent:  $B(x,loc^i^j^k)$ 

%agent: T(s,org~i~j~k,dst~i~j~k)

### Diffusion events

pre- $\kappa$  expander

hurbin

Introductio

What is κ κ syntax

DLab

DLab's curre work

Space-related

Timming contr

#### #Rules

### #A diffusions

$$\begin{split} &A(loc^\text{`}i,x,c),T(org^\text{`}i,dst^\text{`}j) \rightarrow A(loc^\text{`}j,x,c),T(org^\text{`}i,dst^\text{`}j) \ @ 'Adiff_\text{ij'} \\ &A(loc^\text{`}i,x,c),T(org^\text{`}i,dst^\text{`}k) \rightarrow A(loc^\text{`}k,x,c),T(org^\text{`}i,dst^\text{`}k) \ @ 'Adiff_\text{ik'} \\ &A(loc^\text{`}j,x,c),T(org^\text{`}j,dst^\text{`}i) \rightarrow A(loc^\text{`}i,x,c),T(org^\text{`}j,dst^\text{`}i) \ @ 'Adiff_\text{ji'} \\ &A(loc^\text{`}j,x,c),T(org^\text{`}j,dst^\text{`}k) \rightarrow A(loc^\text{`}k,x,c),T(org^\text{`}j,dst^\text{`}k) \ @ 'Adiff_\text{jk'} \\ &A(loc^\text{`}k,x,c),T(org^\text{`}k,dst^\text{`}i) \rightarrow A(loc^\text{`}i,x,c),T(org^\text{`}k,dst^\text{`}i) \ @ 'Adiff_\text{ki'} \\ &A(loc^\text{`}k,x,c),T(org^\text{`}k,dst^\text{`}j) \rightarrow A(loc^\text{`}j,x,c),T(org^\text{`}k,dst^\text{`}j) \ @ 'Adiff_\text{ki'} \\ \end{split}$$

pre- $\kappa$  expander

nurbina

#### Introductio

What is  $\kappa$ 

κ syntax

Nappa at DLab DLab's curr

work

simulations

Timming control

### **#Signatures**

%agent: S(x)

%agent: Z()

%agent: V(p,n)

### #Rules

'Infection' Z(),S(x)  $\rightarrow$  Z(),S(x!1),V(p!1,n) @ 'infection\_rate' 'Polymerization' V(n)  $\rightarrow$  V(n!1),V(p!1,n) @ 'polymer\_rate' 'Expression' S(x!1),V(p!1,n!2),V(p!2,n!3),V(p!3,n!4), \V(p!4,n!5),V(p!5,n!6),V(p!6,n!7),V(p!7,n!8),V(p!8,n!9), \V(p!9,n!10),V(p!10,n)  $\rightarrow$  Z() @ [inf]

```
pre-\kappa expander
```

Hurbina

Introduction

vvnat is κ κ syntax

Kappa at DLab

DLab's currer work

work Space-related

simulations

Timming control pre-Kappa expander

```
\# Signatures
```

%agent: S(x) %agent: Z()

%agent: V(p,n)

### #Rules

'Infection' Z(),S(x)  $\rightarrow$  Z(),S(x!1),V(p!1,n) @ 'infection\_rate' 'Polymerization' V(n)  $\rightarrow$  V(n!1),V(p!1,n) @ 'polymer\_rate' 'Expression' S(x!1),V(p!1,n!2),V(p!2,n!3),V(p!3,n!4), \V(p!4,n!5),V(p!5,n!6),V(p!6,n!7),V(p!7,n!8),V(p!8,n!9), \V(p!9,n!10),V(p!10,n)  $\rightarrow$  Z() @ [inf]

```
pre-\kappa expander
```

nurbina

Introduction

What is  $\kappa$  syntax

Kappa at DLab

DLab's currer work

work Space-related

simulations

Timming control pre-Kappa

```
#Signatures
```

%agent: S(x)

%agent: Z()

%agent: V(p,n)

### #Rules

'Infection'  $Z(),S(x) \rightarrow Z(),S(x!1),V(p!1,n)$  @ 'infection\_rate' 'Polymerization'  $V(n) \rightarrow V(n!1),V(p!1,n)$  @ 'polymer\_rate'

'Expression'  $S(x!1),V(p!1,n!2),V(p!2,n!3),V(p!3,n!4), V(p!4,n!5),V(p!5,n!6),V(p!6,n!7),V(p!7,n!8),V(p!8,n!9), V(p!9,n!10),V(p!10,n) <math>\rightarrow Z()$  @ [inf]

```
pre-\kappa expander
```

hurbin

Introductio What is  $\kappa$ 

κ syntax Kappa at

DLab DLab's currer work

work Space-related

simulations
Timming contr

Timming control pre-Kappa expander

```
\# Signatures
```

%agent: S(x)

%agent: Z()

%agent: V(p,n)

### #Rules

'Infection' Z(),S(x)  $\rightarrow$  Z(),S(x!1),V(p!1,n) @ 'infection\_rate' 'Polymerization' V(n)  $\rightarrow$  V(n!1),V(p!1,n) @ 'polymer\_rate' 'Expression' S(x!1),V(p!1,n!2),V(p!2,n!3),V(p!3,n!4), \ V(p!4,n!5),V(p!5,n!6),V(p!6,n!7),V(p!7,n!8),V(p!8,n!9), \ V(p!9,n!10),V(p!10,n)  $\rightarrow$  Z() @ [inf]

pre- $\kappa$  expander

hurbin

Introduction

κ syntax Kanna at

DLab's currer

pre-Kappa expander

work Space-related simulations Timming contro A Python (V2) script that takes as input a (built in-house) pre- $\kappa$  file and outputs a kappa file which can subsequently be used with KaSim.

This is done using Lexer & Parser techniques, available in Python through ply library.

It facilitates  $\kappa$  abstraction while reducing error-proneness.

Freely available on https://github.com/ajendrex/expander/

pre- $\kappa$  expander

hurbin

Introductio

κ syntax

Nappa at DLab

DLab's curren work Space-related

Space-related simulations Timming contro

pre-Kappa expander A Python (V2) script that takes as input a (built in-house) pre- $\kappa$  file and outputs a kappa file which can subsequently be used with KaSim.

This is done using Lexer & Parser techniques, available in Python through ply library.

It facilitates  $\kappa$  abstraction while reducing error-proneness. Freely available on https://github.com/ajendrex/expander

## pre- $\kappa$ expander

hurbin

### Introductio

Kappa at

#### DLab DLab's currer work

work Space-related simulations Timming contro

pre-Kappa expander A Python (V2) script that takes as input a (built in-house) pre- $\kappa$  file and outputs a kappa file which can subsequently be used with KaSim.

This is done using Lexer & Parser techniques, available in Python through ply library.

It facilitates  $\kappa$  abstraction while reducing error-proneness.

Freely available on https://github.com/ajendrex/expander/

## pre- $\kappa$ expander

hurbin

#### Introductio What is $\kappa$

Kappa at OLah

#### DLab's currer work Space-related

Space-related simulations Timming contri pre-Kappa

expander

A Python (V2) script that takes as input a (built in-house) pre- $\kappa$  file and outputs a kappa file which can subsequently be used with KaSim.

This is done using Lexer & Parser techniques, available in Python through ply library.

It facilitates  $\kappa$  abstraction while reducing error-proneness.

Freely available on https://github.com/ajendrex/expander/

```
pre-\kappa expander
```

hurbin

Introduction

κ syntax

Kappa at DLab

DLab's current work

Space-related simulations Timming control

pre-Kappa expander

```
#Locations
```

%loc: i 100

%loc: j 1000

%loc: k 500

#Location list %locl: all i j k #Signatures

%expand-agent: all A(x,c)%expand-agent: all B(x)

gives:

%agent: A(x,c,loc~i~j~k)

%agent: B(x,loc~i~j~k)

```
pre-\kappa expander
```

hurbin

Introduction

What is κ κ syntax

DLab's curi

DLab's current work Space-related

simulations
Timming contro

Timming contropre-Kappa expander

```
#Locations
```

%loc: i 100

%loc: j 1000 %loc: k 500

/610C. K 300

#Location list %locl: all i j k

#Signatures

%expand-agent: all A(x,c)

%expand-agent: all B(x)

gives:

%agent: A(x,c,loc~i~j~k)

%agent: B(x,loc~i~j~k)

```
pre-\kappa expander
```

hurbin

Introduction What is  $\kappa$ 

Kappa at DLab

DLab's current work Space-related simulations

simulations Timming contro pre-Kappa expander

```
#Locations
```

%loc: i 100

%loc: j 1000

%loc: k 500

#Location list %locl: all i j k

#Signatures

%expand-agent: all A(x,c) %expand-agent: all B(x)

gives:

%agent: A(x,c,loc~i~j~k)

%agent: B(x,loc~i~j~k)

```
pre-\kappa expander
```

hurbin

Introduction What is  $\kappa$ 

Kappa at DLab

DLab's current work Space-related simulations

Timming contro pre-Kappa expander

```
#Locations
```

%loc: i 100

%loc: j 1000 %loc: k 500

7610C: K 500

#Location list %locl: all i j k

#Signatures

%expand-agent: all A(x,c) %expand-agent: all B(x)

gives:

%agent:  $A(x,c,loc^ij^k)$ %agent:  $B(x,loc^ij^k)$ 

```
pre-\kappa
              #Locations
 expander
              %loc: i 100
              %loc: i 1000
              %loc: k 500
              #Location list
              %locl: all i j k
              #Initializations (expand if densities are equal)
              %expand-init: all ADensity A(x,c)
              %expand-init: all BDensity B(x)
pre-Kappa
expander
```

4□ > 4問 > 4 = > 4 = > ■ 900

```
pre-\kappa
             #Locations
 expander
             %loc: i 100
             %loc: i 1000
             %loc: k 500
             #Location list
             %locl: all i j k
             #Initializations (expand if densities are equal)
             %expand-init: all ADensity A(x,c)
             %expand-init: all BDensity B(x)
pre-Kappa
expander
             gives:
             %init: ADensity * 100 A(x,c,loc~i)
             %init: ADensity * 1000 A(x,c,loc~j)
             %init: ADensity * 500 A(x,c,loc~k)
             %init: BDensity * 100 B(x,loc~i)
             %init: BDensity * 1000 B(x,loc~j)
             %init: BDensity * 500 B(x,loc~k)
```

4□ > 4問 > 4 = > 4 = > ■ 900

 $pre-\kappa$ expander

pre-Kappa expander

A bimolecular stochastic rate constant  $\gamma$ , expressed in  $s^{-1}$  molecule<sup>-1</sup>, is related to its deterministic counterpart k, expressed in  $s^{-1}M^{-1}$  as

$$\gamma = \frac{k}{AV},\tag{1}$$

where A is Avogadro's number.

Krivine et. al. Programs as models: Execution. Unpublised work.

```
pre-\kappa
expander
```

pre-Kappa expander

```
#Locations
```

%loc: i 100 %loc: i 1000

%loc: k 500 #Location list %locl: all i j k

```
pre-\kappa
expander
```

pre-Kappa expander

**#Locations** 

%loc: i 100 %loc: i 1000 %loc: k 500

#Location list %locl: all i j k

#A binds B

%expand-rule: all  $A(x),B(x) \rightarrow A(x!1),B(x!1)$  @ 'on\_base\_rate'

```
pre-\kappa expander
```

hurbina

Introductio
What is κ
κ syntax

Kappa at DLab

DLab's current work Space-related

simulations
Timming contr
pre-Kappa
expander

```
#Locations %loc: i 100 %loc: j 1000 %loc: k 500 #Location list %loc!: all i j k #A binds B %expand-rule: all A(x),B(x) \rightarrow A(x!1),B(x!1) @ 'on_base_rate' gives:
```

 $A(x,loc^{\sim}i),B(x,loc^{\sim}i) \rightarrow A(x!1,loc^{\sim}i),B(x!1,loc^{\sim}i)$  @ 'on\_base\_rate' / 100

 $pre-\kappa$ expander

pre-Kappa expander

### #Location matrices

%locm:

TM	i	j	k
i	0	0.5	1.5
j	2.0	0	1.8
k	1.0	1.1	0

gives: 
$$A(loc^{-}i,x,c), T(org^{-}i,dst^{-}j) \rightarrow A(loc^{-}j,x,c), T(org^{-}i,dst^{-}j) @ (1*0.5) / 100 \\ A(loc^{-}i,x,c), T(org^{-}i,dst^{-}k) \rightarrow A(loc^{-}k,x,c), T(org^{-}i,dst^{-}k) @ (1*1.5) / 100 \\ A(loc^{-}j,x,c), T(org^{-}j,dst^{-}i) \rightarrow A(loc^{-}i,x,c), T(org^{-}j,dst^{-}i) @ (1*2.0) / 1000 \\ A(loc^{-}j,x,c), T(org^{-}j,dst^{-}k) \rightarrow A(loc^{-}k,x,c), T(org^{-}j,dst^{-}k) @ (1*1.8) / 100 \\ A(loc^{-}k,x,c), T(org^{-}k,dst^{-}i) \rightarrow A(loc^{-}i,x,c), T(org^{-}k,dst^{-}i) @ (1*1.0) / 500 \\ A(loc^{-}k,x,c), T(org^{-}k,dst^{-}j) \rightarrow A(loc^{-}j,x,c), T(org^{-}k,dst^{-}j) @ (1*1.1) / 500 \\ A(loc^{-}k,x,c), T(org^{-}k,dst^{-}j) \rightarrow A(loc^{-}j,x,c), T(org^{-}k,dst^{-}j) @ (1*1.1) / 500 \\ A(loc^{-}k,x,c), T(org^{-}k,dst^{-}j) \rightarrow A(loc^{-}j,x,c), T(org^{-}k,dst^{-}j) @ (1*1.1) / 500 \\ A(loc^{-}k,x,c), T(org^{-}k,dst^{-}j) \rightarrow A(loc^{-}j,x,c), T(org^{-}k,dst^{-}j) @ (1*1.1) / 500 \\ A(loc^{-}k,x,c), T(org^{-}k,dst^{-}j) \rightarrow A(loc^{-}j,x,c), T(org^{-}k,dst^{-}j) @ (1*1.1) / 500 \\ A(loc^{-}k,x,c), T(org^{-}k,dst^{-}j) \rightarrow A(loc^{-}k,x,c), T(org^{-}k,dst^{-}j) @ (1*1.1) / 500 \\ A(loc^{-}k,x,c), T(org^{-}k,dst^{-}k) \rightarrow A(loc^{-}k,x,c), T(org^{-}k,dst^{-}k) & A(loc^{-}k,x,c), T($$

pre- $\kappa$  expander

hurbin

Introductio What is  $\kappa$ 

Kappa at DLab

DLab's curren

Space-related simulations

Simulations
Timming contr
pre-Kappa
expander

```
#Location matrices
```

%locm:

TM	i	j	k
i	0	0.5	1.5
j	2.0	0	1.8
k	1.0	1.1	0

### #A diffusions

%expand-rule: TM  $A(x,c),T() \rightarrow A(\%,x,c),T() @ 1$ 

#### $pre-\kappa$ expander

pre-Kappa expander

### #Location matrices

#### %locm:

```
TM
                    k
            0.5
       0
                   15
      20
             0
                   18
      1.0
             1.1
```

### #A diffusions

%expand-rule: TM A(x,c),T()  $\rightarrow$  A(%,x,c),T() @ 1

$$\begin{split} & A(loc^{\sim}i,x,c), T(org^{\sim}i,dst^{\sim}j) \rightarrow A(loc^{\sim}j,x,c), T(org^{\sim}i,dst^{\sim}j) \ @ \ (1 * 0.5) \ / \ 100 \\ & A(loc^{\sim}i,x,c), T(org^{\sim}i,dst^{\sim}k) \rightarrow A(loc^{\sim}k,x,c), T(org^{\sim}i,dst^{\sim}k) \ @ \ (1 * 1.5) \ / \ 100 \\ & A(loc^{\sim}j,x,c), T(org^{\sim}j,dst^{\sim}i) \rightarrow A(loc^{\sim}i,x,c), T(org^{\sim}j,dst^{\sim}i) \ @ \ (1 * 2.0) \ / \ 1000 \\ & A(loc^{\sim}j,x,c), T(org^{\sim}j,dst^{\sim}k) \rightarrow A(loc^{\sim}k,x,c), T(org^{\sim}j,dst^{\sim}k) \ @ \ (1 * 1.8) \ / \ 1000 \\ & A(loc^{\sim}k,x,c), T(org^{\sim}k,dst^{\sim}i) \rightarrow A(loc^{\sim}i,x,c), T(org^{\sim}k,dst^{\sim}i) \ @ \ (1 * 1.0) \ / \ 500 \\ & A(loc^{\sim}k,x,c), T(org^{\sim}k,dst^{\sim}j) \rightarrow A(loc^{\sim}j,x,c), T(org^{\sim}k,dst^{\sim}j) \ @ \ (1 * 1.1) \ / \ 500 \\ \end{split}$$

```
pre-\kappa expander hurbina troduction what is \kappa
```

What is  $\kappa$   $\kappa$  syntax

DLab DLab's curre

DLab's curren work

Space-related simulations

Timming contro pre-Kappa expander

```
#Location matrices
```

### %locm:

```
TM i j k
i 0 0.5 1.5
j 2.0 0 1.8
k 1.0 1.1 0
```

### **#Observing transporters**

%expand-obs: TM 'Transporter(%org,%dst)' T()

```
%obs: 'Transporter(i,j)' T(org~i,dst~j)
%obs: 'Transporter(i,k)' T(org~i,dst~k)
%obs: 'Transporter(j,i)' T(org~j,dst~i)
%obs: 'Transporter(j,k)' T(org~j,dst~k]
%obs: 'Transporter(k,i)' T(org~k,dst~i)
```

```
pre-\kappa
expander
```

pre-Kappa expander

```
#Location matrices
```

#### %locm:

```
TM
            0.5
                   15
       0
      2.0
             0
                   18
      1.0
            1.1
```

### **#Observing transporters**

%expand-obs: TM 'Transporter(%org,%dst)' T()

```
%obs: 'Transporter(i,j)' T(org~i,dst~j)
```

%obs: 'Transporter(i,k)' 
$$T(org^i,dst^k)$$

%obs: 'Transporter(j,i)' 
$$T(org^{j},dst^{i})$$

%obs: 'Transporter(j,k)' 
$$T(org^{j},dst^{k})$$

%obs: 'Transporter(k,j)' 
$$T(org^k,dst^j)$$

# pre-Kappa syntax: Chains

pre- $\kappa$  expander

hurbin

#### Introductio

What is i κ syntax

### Kappa at

DLab's current work

Space-related simulations Fimming control

pre-Kappa expander

### #Rules

 $\text{`Expression' S(x!1),V(p!1,n!2),V(p!2,n!3),...,V(p!10,n)} \rightarrow \text{Z() @ [inf]}$ 

### gives:

'Expression'  $S(x!1),V(p!1,n!2),V(p!2,n!3),V(p!3,n!4), V(p!4,n!5),V(p!5,n!6),V(p!6,n!7),V(p!7,n!8),V(p!8,n!9), V(p!9,n!10),V(p!10,n) <math>\rightarrow Z()$  @ [inf]

# pre-Kappa syntax: Chains

pre- $\kappa$  expander

nurbin

### Introductio

κ syntax

### Kappa at

expander

DLab's current work Space-related simulations Timming control pre-Kappa

### #Rules

 $\text{`Expression' S(x!1),V(p!1,n!2),V(p!2,n!3),...,V(p!10,n)} \rightarrow \text{Z() @ [inf]}$ 

### gives:

'Expression'  $S(x|1),V(p|1,n|2),V(p|2,n|3),V(p|3,n|4), V(p|4,n|5),V(p|5,n|6),V(p|6,n|7),V(p|7,n|8),V(p|8,n|9), V(p|9,n|10),V(p|10,n) <math>\rightarrow$  Z() @ [inf]