

Event modelling of message interchange in stochastic neural ensembles

Vicenç Gomez Andreas Kaltenbrunner Vicente López

Departament de Tecnologia
Universitat Pompeu Fabra

July 16th, 2006 / IJCNN'06

Outline

Introduction

Mesoscopic event-driven modeling

A case study: discrete and stochastic ensembles

Extension to continuous time models

Simulation results

Conclusions

Introduction

- ▶ Often very detailed models are used for the study of **collective dynamics of networks of spiking neurons**
- ▶ State of a neuron i is characterized by its membrane potential v_i (a **microscopic** magnitude)

$$\dot{v}_i = f(v_1, \dots, \dots, v_N)$$

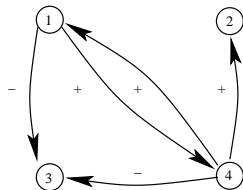
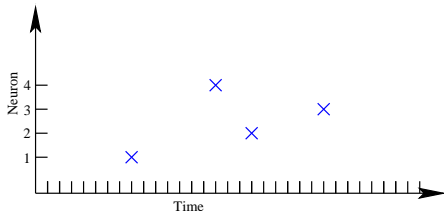
- ▶ Spike trains $\langle T_k^i \rangle$ are analyzed (a **mesoscopic** magnitude)

We propose a mesoscopic modeling technique which does not determine the microstates

Existing event-driven approaches

The membrane potential drives the simulation

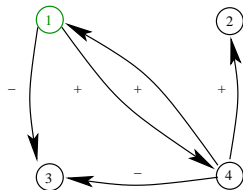
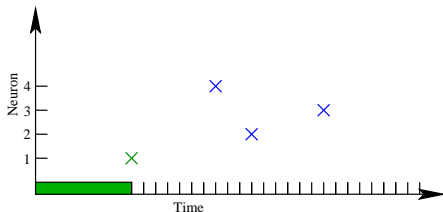
- ▶ Spikes are stereotypical events.
- ▶ Computation time between events (spikes) is prevented.
- ▶ **Stochastic** dynamics: *Reutimann et al, 2003*
- ▶ Event-driven simulation loop:
 1. Find next event to be processed.
 2. Update states of the affected neurons.
 3. Schedule all possible generated events.



Existing event-driven approaches

The membrane potential drives the simulation

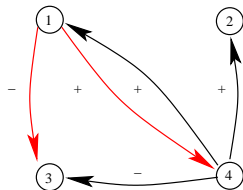
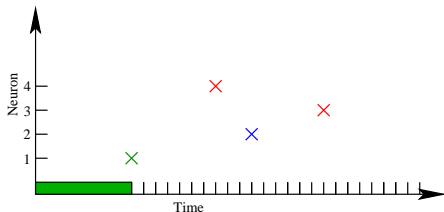
- ▶ Spikes are stereotypical events.
- ▶ Computation time between events (spikes) is prevented.
- ▶ **Stochastic** dynamics: *Reutimann et al, 2003*
- ▶ Event-driven simulation loop:
 1. **Find next event to be processed.**
 2. Update states of the affected neurons.
 3. Schedule all possible generated events.



Existing event-driven approaches

The membrane potential drives the simulation

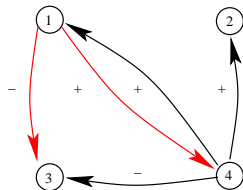
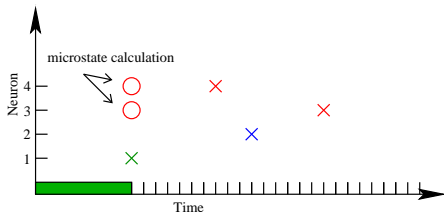
- ▶ Spikes are stereotypical events.
- ▶ Computation time between events (spikes) is prevented.
- ▶ **Stochastic** dynamics: *Reutimann et al, 2003*
- ▶ Event-driven simulation loop:
 1. Find next event to be processed.
 2. **Update states of the affected neurons.**
 3. Schedule all possible generated events.



Existing event-driven approaches

The membrane potential drives the simulation

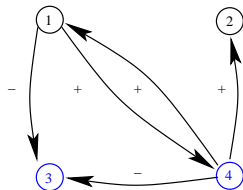
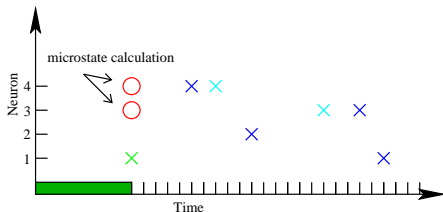
- ▶ Spikes are stereotypical events.
- ▶ Computation time between events (spikes) is prevented.
- ▶ **Stochastic** dynamics: *Reutimann et al, 2003*
- ▶ Event-driven simulation loop:
 1. Find next event to be processed.
 2. **Update states of the affected neurons.**
 3. Schedule all possible generated events.



Existing event-driven approaches

The membrane potential drives the simulation

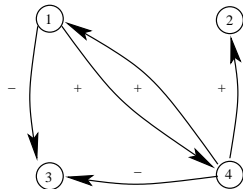
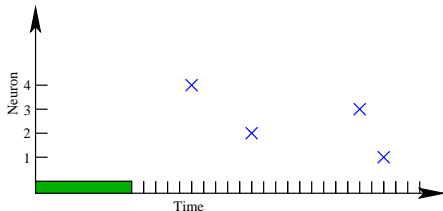
- ▶ Spikes are stereotypical events.
- ▶ Computation time between events (spikes) is prevented.
- ▶ **Stochastic** dynamics: *Reutimann et al, 2003*
- ▶ Event-driven simulation loop:
 1. Find next event to be processed.
 2. Update states of the affected neurons.
 3. **Schedule all possible generated events.**



Existing event-driven approaches

The membrane potential drives the simulation

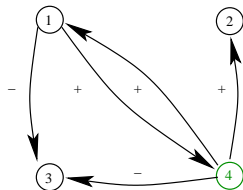
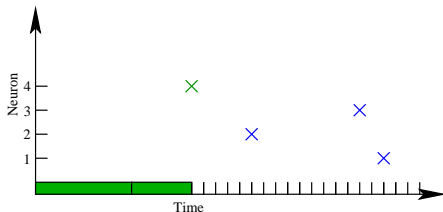
- ▶ Spikes are stereotypical events.
- ▶ Computation time between events (spikes) is prevented.
- ▶ **Stochastic** dynamics: *Reutimann et al, 2003*
- ▶ Event-driven simulation loop:
 1. Find next event to be processed.
 2. Update states of the affected neurons.
 3. Schedule all possible generated events.



Existing event-driven approaches

The membrane potential drives the simulation

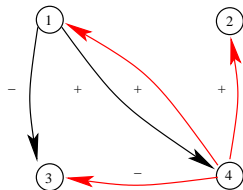
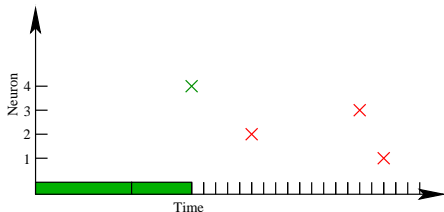
- ▶ Spikes are stereotypical events.
- ▶ Computation time between events (spikes) is prevented.
- ▶ **Stochastic** dynamics: *Reutimann et al, 2003*
- ▶ Event-driven simulation loop:
 1. **Find next event to be processed.**
 2. Update states of the affected neurons.
 3. Schedule all possible generated events.



Existing event-driven approaches

The membrane potential drives the simulation

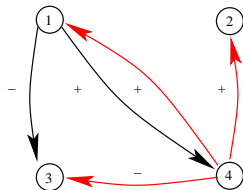
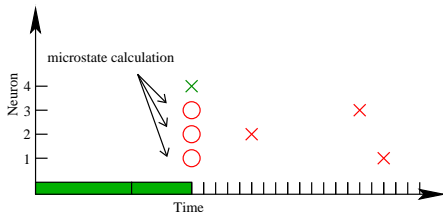
- ▶ Spikes are stereotypical events.
- ▶ Computation time between events (spikes) is prevented.
- ▶ **Stochastic** dynamics: *Reutimann et al, 2003*
- ▶ Event-driven simulation loop:
 1. Find next event to be processed.
 2. **Update states of the affected neurons.**
 3. Schedule all possible generated events.



Existing event-driven approaches

The membrane potential drives the simulation

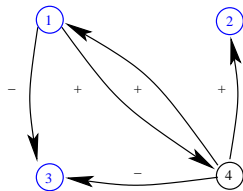
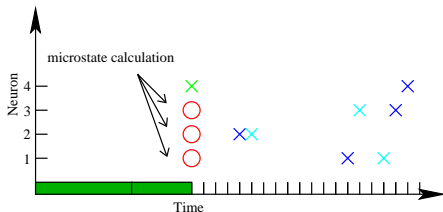
- ▶ Spikes are stereotypical events.
- ▶ Computation time between events (spikes) is prevented.
- ▶ **Stochastic** dynamics: *Reutimann et al, 2003*
- ▶ Event-driven simulation loop:
 1. Find next event to be processed.
 2. **Update states of the affected neurons.**
 3. Schedule all possible generated events.



Existing event-driven approaches

The membrane potential drives the simulation

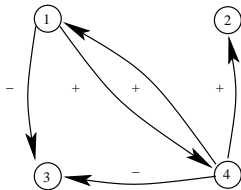
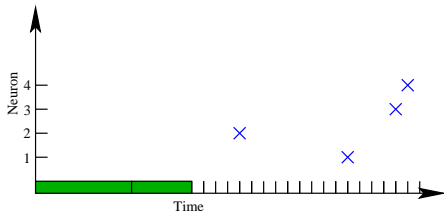
- ▶ Spikes are stereotypical events.
- ▶ Computation time between events (spikes) is prevented.
- ▶ **Stochastic** dynamics: *Reutimann et al, 2003*
- ▶ Event-driven simulation loop:
 1. Find next event to be processed.
 2. Update states of the affected neurons.
 3. **Schedule all possible generated events.**



Existing event-driven approaches

The membrane potential drives the simulation

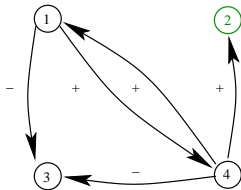
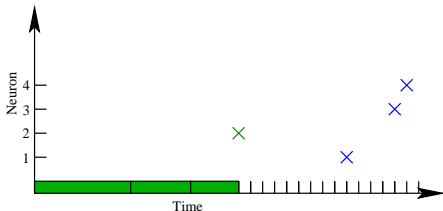
- ▶ Spikes are stereotypical events.
- ▶ Computation time between events (spikes) is prevented.
- ▶ **Stochastic** dynamics: *Reutimann et al, 2003*
- ▶ Event-driven simulation loop:
 1. Find next event to be processed.
 2. Update states of the affected neurons.
 3. Schedule all possible generated events.



Existing event-driven approaches

The membrane potential drives the simulation

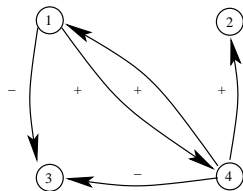
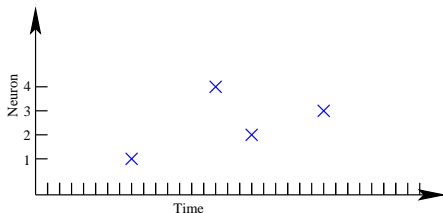
- ▶ Spikes are stereotypical events.
- ▶ Computation time between events (spikes) is prevented.
- ▶ **Stochastic** dynamics: *Reutimann et al, 2003*
- ▶ Event-driven simulation loop:
 1. **Find next event to be processed.**
 2. Update states of the affected neurons.
 3. Schedule all possible generated events.



Proposed **mesoscopic** event-driven

The spike production drives the simulation

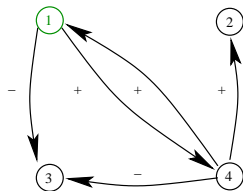
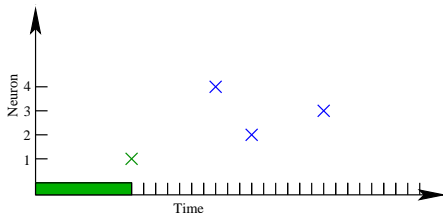
- ▶ New state b_i : next predicted firing time of unit i
- ▶ Predictions are **not discarded** $P(b'_i|b_i) = \frac{P(b_i|b'_i)P(b'_i)}{P(b_i)}$
- ▶ Event-driven simulation loop:
 1. Find next event to be processed.
 2. Update states of the affected neurons.
 3. Schedule all possible generated events.



Proposed **mesoscopic** event-driven

The spike production drives the simulation

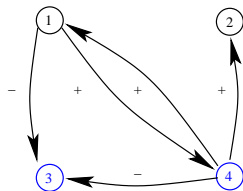
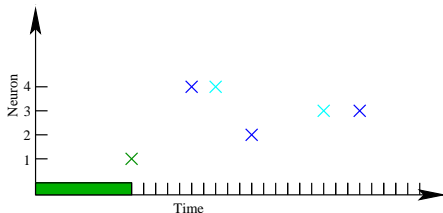
- ▶ New state b_i : next predicted firing time of unit i
- ▶ Predictions are **not discarded** $P(b'_i|b_i) = \frac{P(b_i|b'_i)P(b'_i)}{P(b_i)}$
- ▶ Event-driven simulation loop:
 1. Find next event to be processed.
 2. Update states of the affected neurons.
 3. Schedule all possible generated events.



Proposed **mesoscopic** event-driven

The spike production drives the simulation

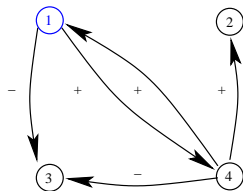
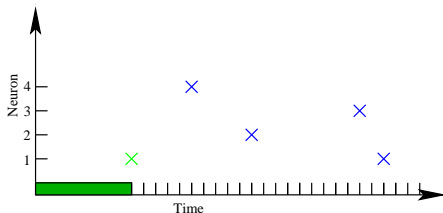
- ▶ New state b_i : next predicted firing time of unit i
- ▶ Predictions are **not discarded** $P(b'_i|b_i) = \frac{P(b_i|b'_i)P(b'_i)}{P(b_i)}$
- ▶ Event-driven simulation loop:
 1. Find next event to be processed.
 2. **Update states of the affected neurons.**
 3. Schedule all possible generated events.



Proposed **mesoscopic** event-driven

The spike production drives the simulation

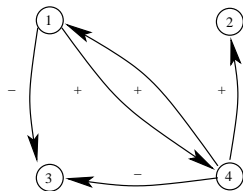
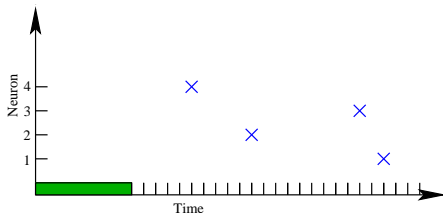
- ▶ New state b_i : next predicted firing time of unit i
- ▶ Predictions are **not discarded** $P(b'_i|b_i) = \frac{P(b_i|b'_i)P(b'_i)}{P(b_i)}$
- ▶ Event-driven simulation loop:
 1. Find next event to be processed.
 2. Update states of the affected neurons.
 3. **Schedule all possible generated events.**



Proposed **mesoscopic** event-driven

The spike production drives the simulation

- ▶ New state b_i : next predicted firing time of unit i
- ▶ Predictions are **not discarded** $P(b'_i|b_i) = \frac{P(b_i|b'_i)P(b'_i)}{P(b_i)}$
- ▶ Event-driven simulation loop:
 1. Find next event to be processed.
 2. Update states of the affected neurons.
 3. Schedule all possible generated events.

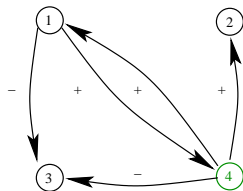
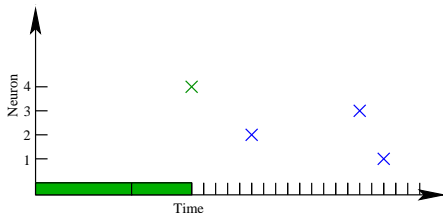


Proposed **mesoscopic** event-driven

The spike production drives the simulation

- ▶ New state b_i : next predicted firing time of unit i
- ▶ Predictions are **not discarded** $P(b'_i|b_i) = \frac{P(b_i|b'_i)P(b'_i)}{P(b_i)}$
- ▶ Event-driven simulation loop:

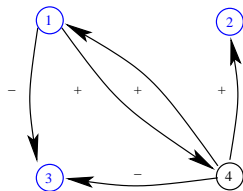
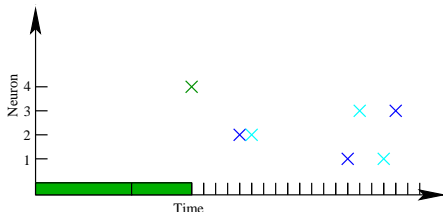
1. Find next event to be processed.
2. Update states of the affected neurons.
3. Schedule all possible generated events.



Proposed **mesoscopic** event-driven

The spike production drives the simulation

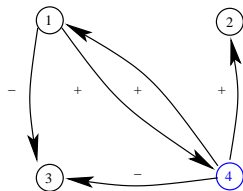
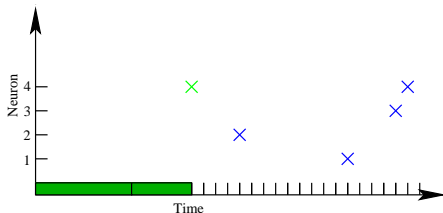
- ▶ New state b_i : next predicted firing time of unit i
- ▶ Predictions are **not discarded**
$$P(b'_i|b_i) = \frac{P(b_i|b'_i)P(b'_i)}{P(b_i)}$$
- ▶ Event-driven simulation loop:
 1. Find next event to be processed.
 2. **Update states of the affected neurons.**
 3. Schedule all possible generated events.



Proposed **mesoscopic** event-driven

The spike production drives the simulation

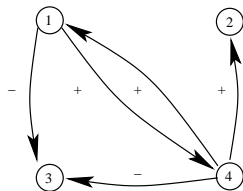
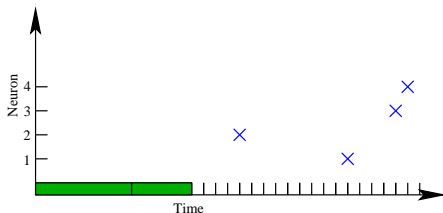
- ▶ New state b_i : next predicted firing time of unit i
- ▶ Predictions are **not discarded** $P(b'_i|b_i) = \frac{P(b_i|b'_i)P(b'_i)}{P(b_i)}$
- ▶ Event-driven simulation loop:
 1. Find next event to be processed.
 2. Update states of the affected neurons.
 3. **Schedule all possible generated events.**



Proposed **mesoscopic** event-driven

The spike production drives the simulation

- ▶ New state b_i : next predicted firing time of unit i
- ▶ Predictions are **not discarded** $P(b'_i|b_i) = \frac{P(b_i|b'_i)P(b'_i)}{P(b_i)}$
- ▶ Event-driven simulation loop:
 1. Find next event to be processed.
 2. Update states of the affected neurons.
 3. Schedule all possible generated events.

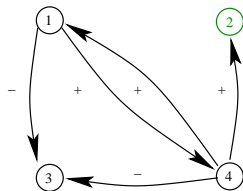
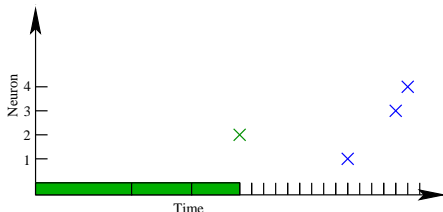


Proposed **mesoscopic** event-driven

The spike production drives the simulation

- ▶ New state b_i : next predicted firing time of unit i
- ▶ Predictions are **not discarded** $P(b'_i|b_i) = \frac{P(b_i|b'_i)P(b'_i)}{P(b_i)}$
- ▶ Event-driven simulation loop:

1. Find next event to be processed.
2. Update states of the affected neurons.
3. Schedule all possible generated events.



A case study: Discrete and stochastic integrate-and-fire ensembles

The microscopic model

Subthreshold dynamics:

$$v_i(t+1) - v_i(t) = -\mu + \Delta I_{noise}(t) + I_{rec}(t - t_{delay})$$

where

μ leaky term	\rightarrow	0
I_{noise} stochastic process	\rightarrow	Bernoulli p
$I_{rec}(t)$ induced activity	\rightarrow	$\sum_{j \neq i} \epsilon_{ij} \Theta_j^L(v_j(t))$

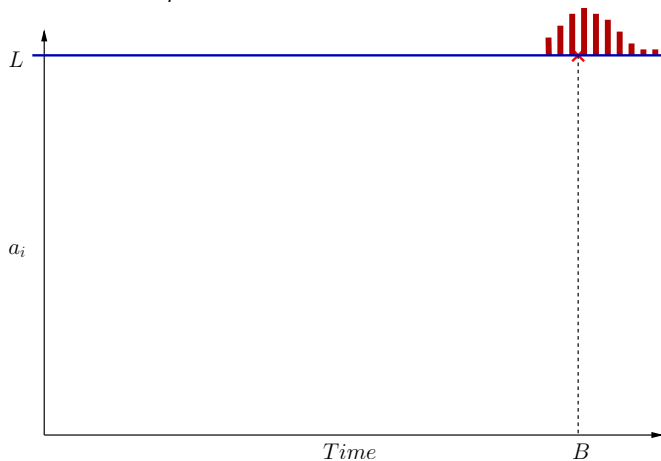
If threshold L is reached then $v_i(t_{spike} + t_{refr}) = V_R$

Example: Neural ensembles

Mesoscopic reformulation

Belief updating for unit i

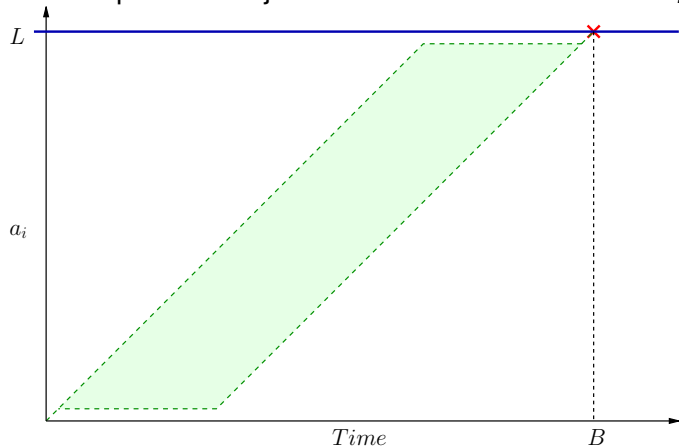
First belief b_j follows the unconditional ISI distribution



Mesoscopic reformulation

Belief updating for unit i

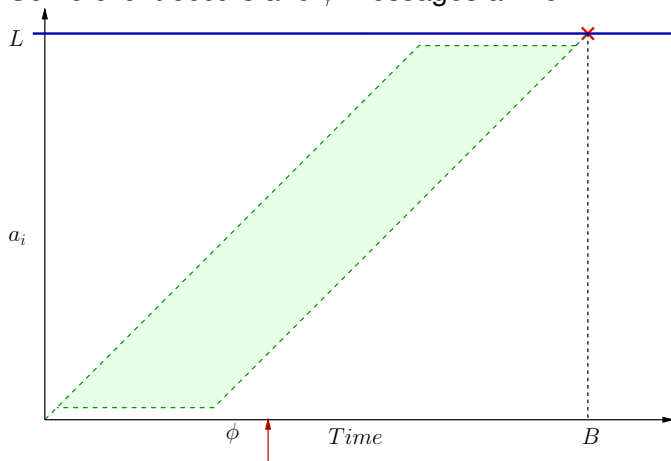
A set of possible trajectories is defined from t until b_i



Mesoscopic reformulation

Belief updating for unit i

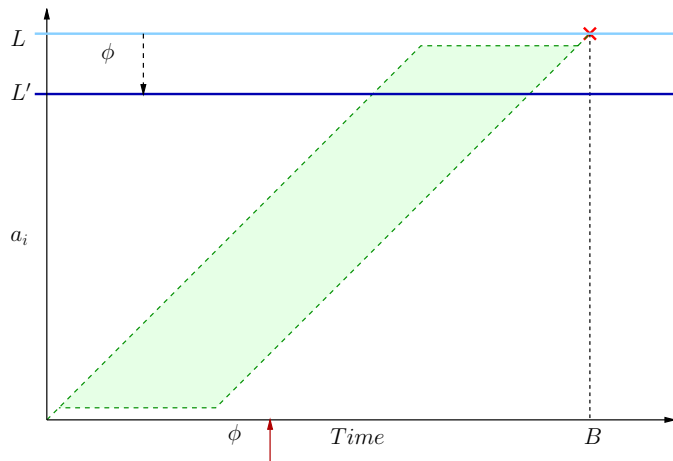
Some event occurs and ϕ messages arrive



Mesoscopic reformulation

Belief updating for unit i

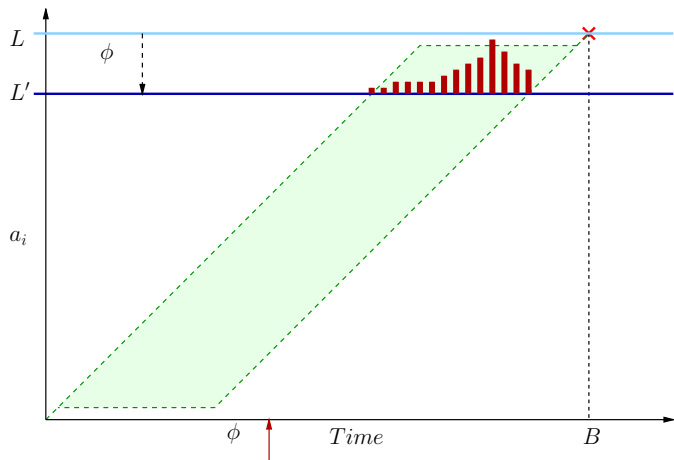
► Lowering of the threshold L



Mesoscopic reformulation

Belief updating for unit i

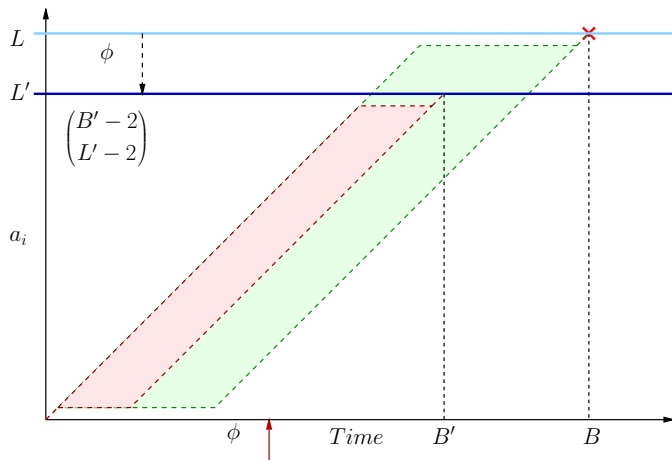
- ▶ A conditional ISI probability distribution is defined for b'



Mesoscopic reformulation

Belief updating for unit i

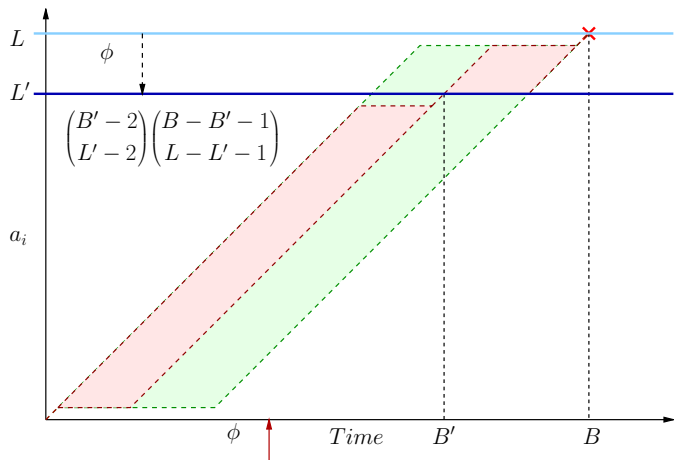
► First set of trajectories



Mesoscopic reformulation

Belief updating for unit i

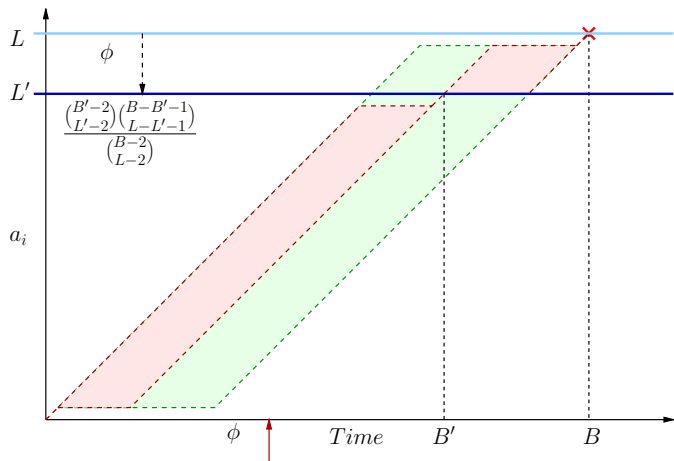
► Second set of trajectories



Mesoscopic reformulation

Belief updating for unit i

► Final conditional ISI probability distribution for b'



Mesoscopic reformulation

First-Passage-Time probability distributions (discrete time)

Unconditional ISI probability distribution

Negative Binomial distribution:

$$P_{NB}(b_i) = \binom{b_i - 1}{L_i - 2} p^{L_i - 1} (1 - p)^{b_i - L_i + 1} \quad \text{for } b_i = 1, 2, \dots$$

Conditional ISI probability distribution

Negative Hypergeometric distribution:

$$P_{NH}(b'_i | b_i) = \frac{P(b_i | b'_i) P(b'_i)}{P(b_i)} = \frac{\binom{b_i - b'_i - 1}{L_i - L'_i - 1} \binom{b'_i - 1}{L'_i - 2}}{\binom{b_i - 1}{L_i - 2}}$$

Mesoscopic reformulation

First-Passage-Time density functions (continuous time)

Unconditional ISI density function

Gamma density function with $\alpha = L - 1$, $\beta = p$:

$$P_{\Gamma}(b_i) = \frac{\beta^{\alpha}}{\Gamma(\alpha)} b_i^{\alpha-1} e^{-\beta b_i} \quad \text{for } b_i \geq 0$$

Conditional ISI density function

Beta density function with $\alpha = L'_i - 1$, $\beta = L_i - L'_i$:

$$P_{\beta}(b'_i | b_i) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \left(1 - \frac{b'_i}{b_i}\right)^{\beta-1} \left(\frac{b'_i}{b_i}\right)^{\alpha-1} \frac{1}{b_i}$$

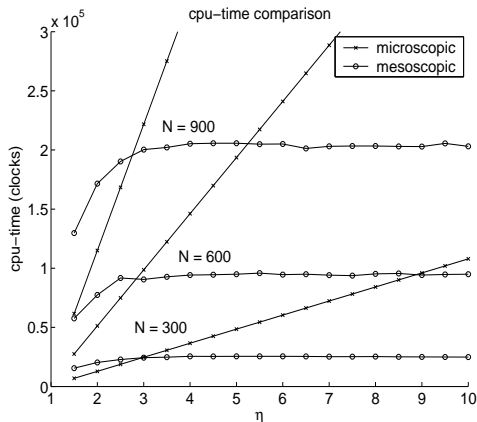
Simulation results

computational complexity

Given τ the interspike period:

$$\text{cost}_{\text{micro}} = \mathcal{O}(\tau N)$$

$$\text{cost}_{\text{meso}} = \mathcal{O}(N^2)$$



Conclusions

A **mesoscopic** framework has been presented:

- ▶ Based on the **spike production and reception**.
- ▶ **Microscopic** dynamics do not enter the model.
- ▶ State updating is defined as an **inference** procedure.
- ▶ Suitable framework when:
 - ▶ Large scale simulations
 - ▶ microstate evolution is not required