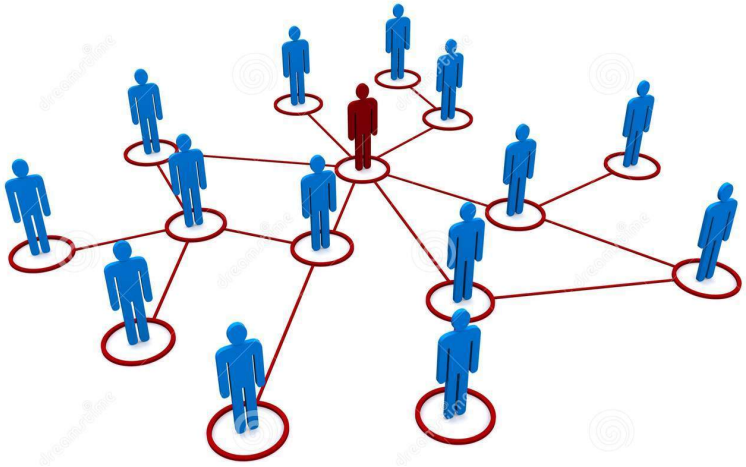
Approximations for dynamics on graphs

Nicolas Gast (Inria, Grenoble)

joint work with Benny Van Houdt (Univ. Antwerp) and Sebastian Allmeier (Inria)

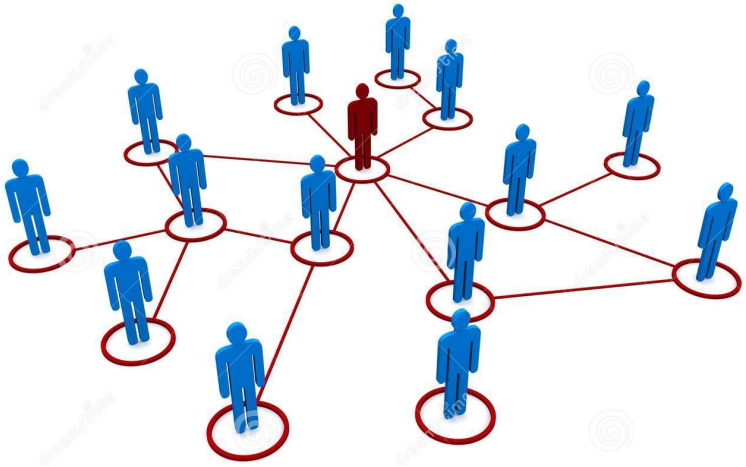
Paris, 8 Days on Network Mathematics, September 2023

Motivation: agents interacting on a graph



- N agents (*e.g.*, servers, neurons, infected people)
- Steady-state properties (*e.g.*, queue lengths, activation, % infected)

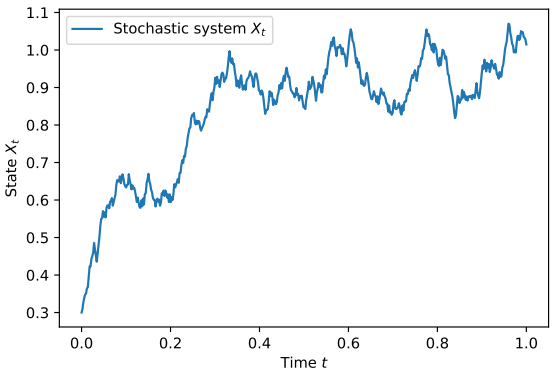
Motivation: agents interacting on a graph

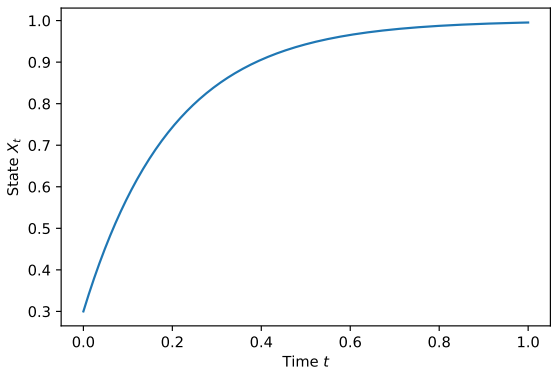


- N agents (*e.g.*, servers, neurons, infected people)
- Steady-state properties (*e.g.*, queue lengths, activation, % infected)

"Theorem": If the graph is dense:

$$\lim_{N\to\infty} \Big(\quad \Big) =$$





ODE (mean field) is asymptotically exact

What can we do for sparse graphs?

- N agents.
- $O(1)$ neighbors per node.

Open questions:

- Tractable and accurate approximations?
- Can we prove anything?

What can we do for sparse graphs?

- N agents.
- $O(1)$ neighbors per node.

Open questions:

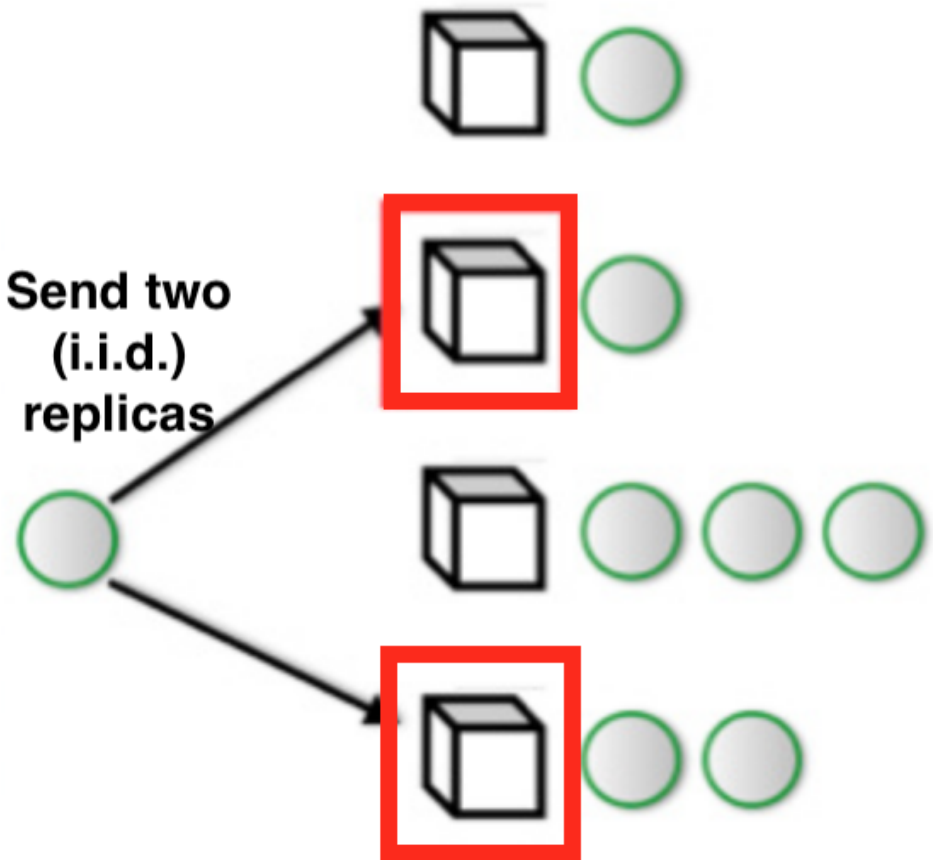
- Tractable and accurate approximations?
- Can we prove anything?

I will use two examples:

- Dynamic random graph: pair/triplet-approximation.
- Information propagation with interference: refined replica.

Outline

Motivation: parrallel systems with redundancy

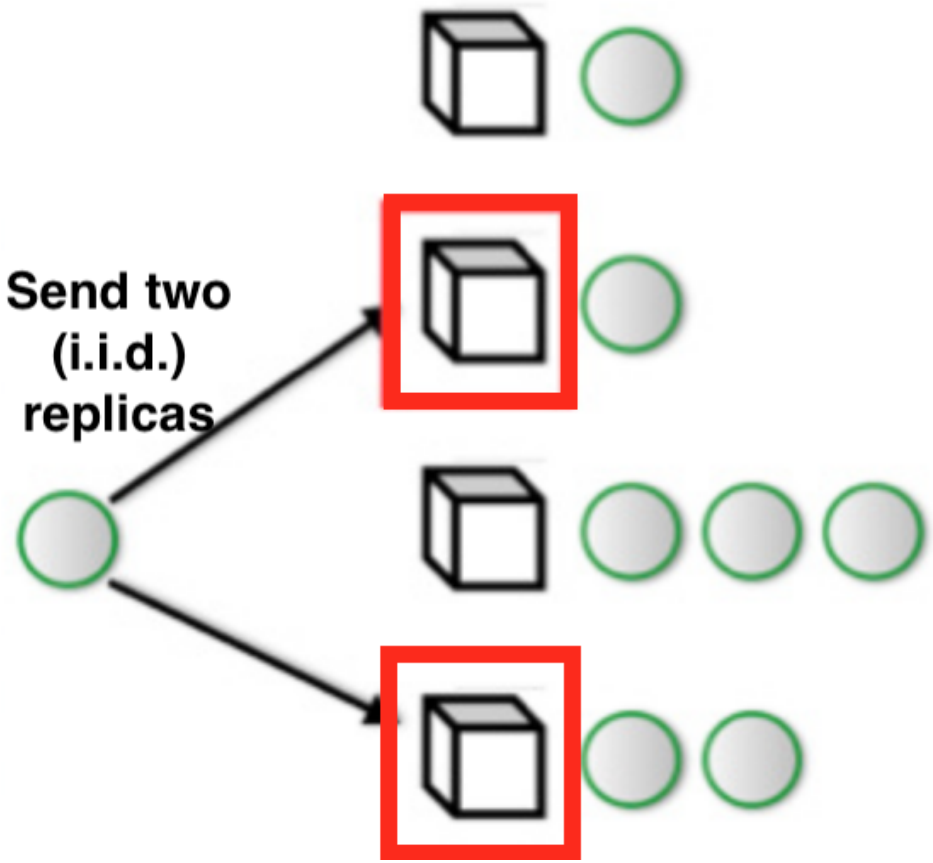


"Simplest" setting:

- Poisson arrival $N\lambda$
- Independent replicas
- Exponential service 1
- Cancel on complete

We want to characterize the queue length distribution.

Motivation: parrallel systems with redundancy



"Simplest" setting:

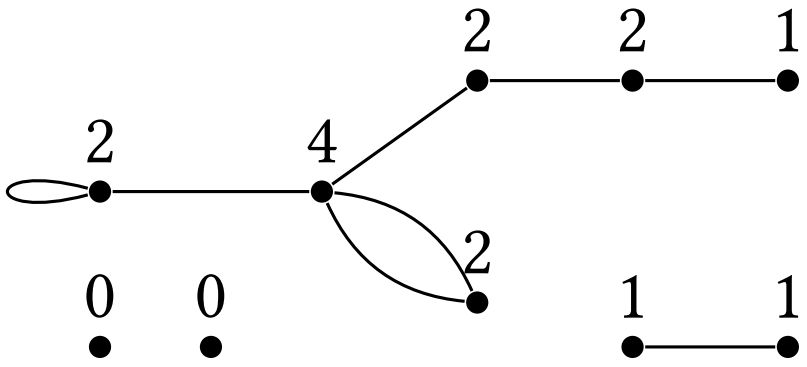
- Poisson arrival $N\lambda$
- Independent replicas
- Exponential service 1
- Cancel on complete

We want to characterize the queue length distribution.

- With FCFS: "easy" (order-independent queues)
- With PS: ??? (only stability is known, $\lambda < 1$)

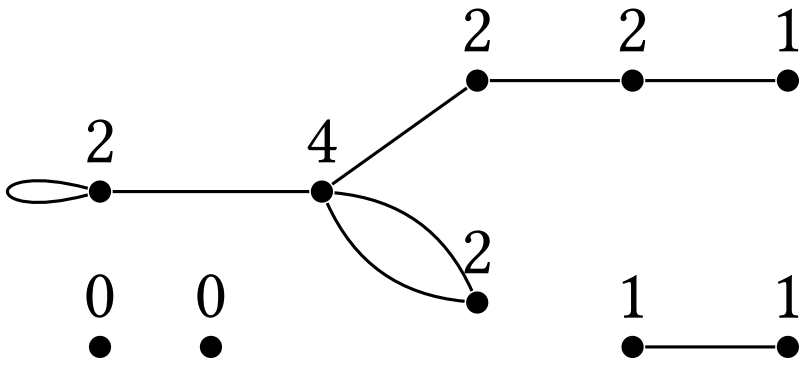
Markovian representation: dynamic graph model

For each job shared by i and j, we add an edge (i, j):



Markovian representation: dynamic graph model

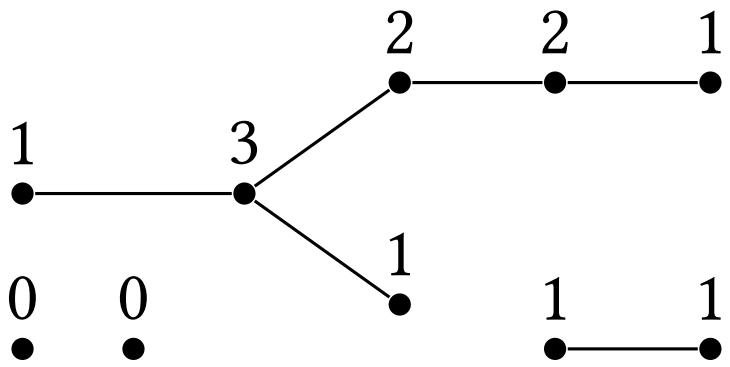
For each job shared by i and j, we add an edge (i, j):



- Each edge is created at rate $2\lambda/N$
- Each node deletes one of its edge at rate 1.

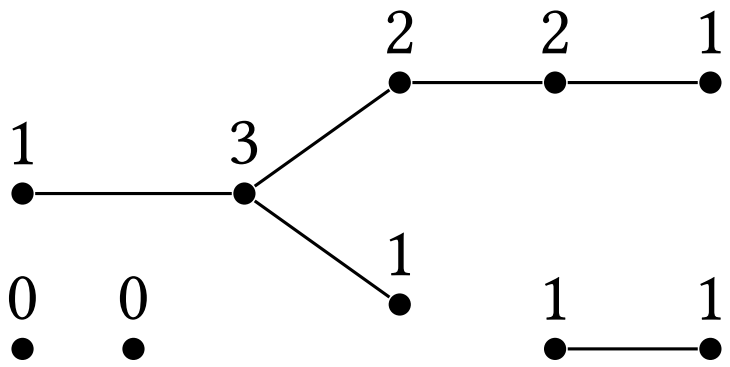
We want to study the degree distribution (=queue length) for large N.

What approximations can we construct?



The graph shoud be simple and locally a tree (because N is large)

What approximations can we construct?

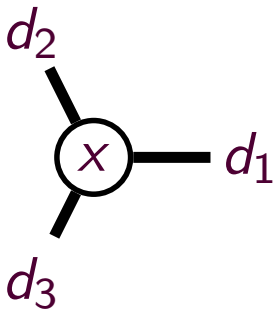


The graph shoud be simple and locally a tree (because N is large)

Approximations:

- Nodes are independent (= mean field)
- Nodes only depend on neighbors (= pair-approximation)

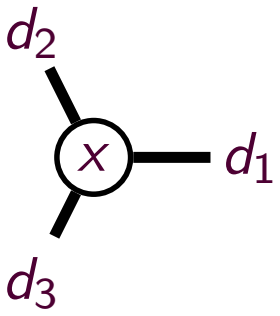
Construction of the mean field approximation



If a node has degree x:

- $x \mapsto x+1$ at rate 2λ
- $x \mapsto x-1$ at rate $1 + \sum_{i=1}^{x} \frac{1}{d_i}$, where d_i is the degree of the ith neighboor.

Construction of the mean field approximation

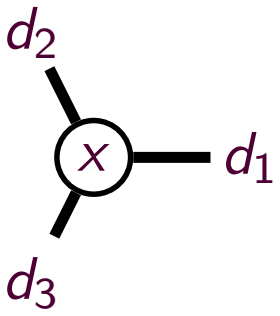


If a node has degree x:

- $x \mapsto x+1$ at rate 2λ
- $x \mapsto x-1$ at rate $1 + \sum_{i=1}^{x} \frac{1}{d_i}$, where d_i is the degree of the ith neighboor.

$$\mathbb{E}\left[\frac{1}{d_i}\right] = ?$$

Construction of the mean field approximation



If a node has degree x:

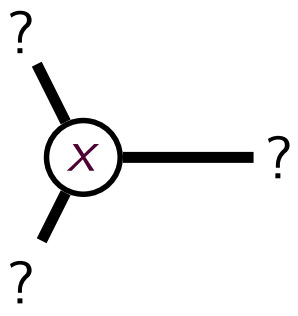
- $x \mapsto x+1$ at rate 2λ
- $x \mapsto x-1$ at rate $1 + \sum_{i=1}^{x} \frac{1}{d_i}$, where d_i is the degree of the ith neighboor.

$$\mathbb{E}\left[\frac{1}{d_i}\right] = \sum_{q \geq 1} \underbrace{\mathbf{P}\left[d_i = q\right]}_{\approx \frac{q\mathbf{P}[\text{degree}=q]}{\bar{q}} \text{ (mean field approximation)}} \qquad \frac{1}{q} = \frac{1-q_0}{\bar{q}},$$

where $\bar{q} = \sum_q q\mathbf{P}\left[\text{degree} = q\right]$ is the average queue length of a node taken at random.

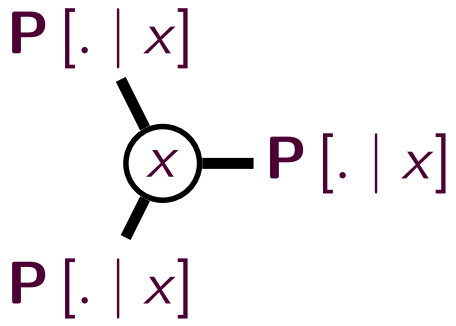
Pair-approximation

Degree of neighboors are not independent.

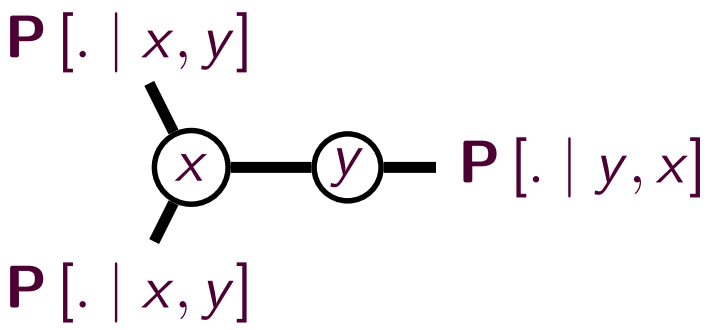


Pair-approximation

Degree of neighboors are not independent.



Pair-approximation



Degree of neighboors are not independent.

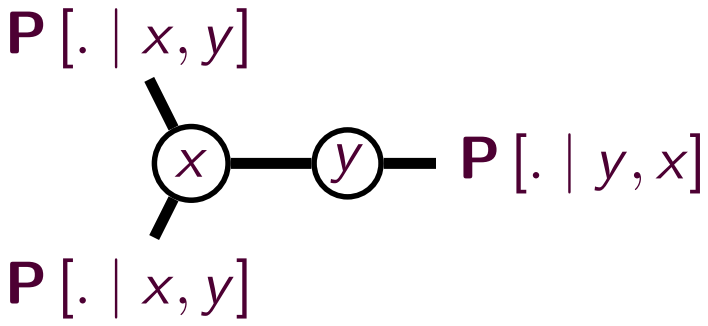
We need to track the proportion of pairs:

$$\pi(x, y) = \frac{1}{N} \#\{\text{connected pairs } (x, y)\}.$$

Approx: $\mathbf{P}[z|x, y] \approx \mathbf{P}[z|x] = \dfrac{\pi(x, z)}{\sum_{z'} \pi(x, z')}$.

Pair-approximation

Degree of neighboors are not independent.



We need to track the proportion of pairs:

$$\pi(x, y) = \frac{1}{N} \#\{\text{connected pairs } (x, y)\}.$$

Approx: $\mathbf{P}[z|x, y] \approx \mathbf{P}[z|x] = \dfrac{\pi(x, z)}{\sum_{z'} \pi(x, z')}$.

If we zoom on one pair:

- Creation of pairs / destruction
- $(x, y) \mapsto (x + 1, y)$: creation of a new neighboor of x
- $(x, y) \mapsto (x - 1, y)$: departure of one of the $x - 1$ neighboors of x.

Can we do triplet approximations?

Yes but... The complexity is large (computation + construction).

Are these approximations just approximations?

Example with $\lambda = 0.7$ and $N = 10^6$ servers.

	q_0	q_1	q_2	q_3	q_4	q_5	q_6
Simu	0.29999	0.28283	0.20311	0.11820	0.05791	0.02451	0.00914

Are these approximations just approximations?

Example with $\lambda = 0.7$ and $N = 10^6$ servers.

	q_0	q_1	q_2	q_3	q_4	q_5	q_6
Simu	0.29999	0.28283	0.20311	0.11820	0.05791	0.02451	0.00914
m-f	0.30000	0.28471	0.20436	0.11795	0.05693	0.02361	0.00858

Are these approximations just approximations?

Example with $\lambda = 0.7$ and $N = 10^6$ servers.

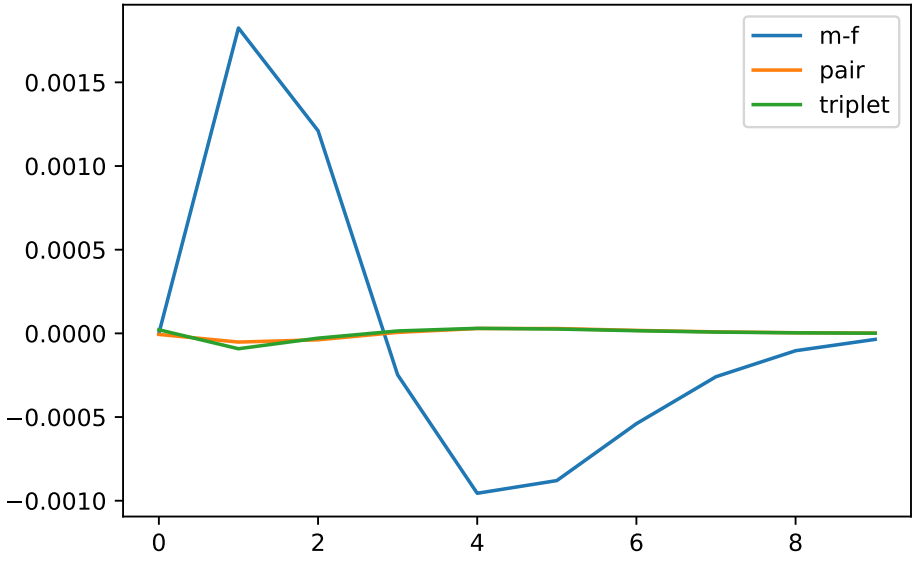
	q_0	q_1	q_2	q_3	q_4	q_5	q_6
Simu	0.29999	0.28283	0.20311	0.11820	0.05791	0.02451	0.00914
m-f	0.30000	0.28471	0.20436	0.11795	0.05693	0.02361	0.00858
Pair	0.30000	0.28288	0.20315	0.11820	0.05788	0.02449	0.00912

Are these approximations just approximations?

Example with $\lambda = 0.7$ and $N = 10^6$ servers.

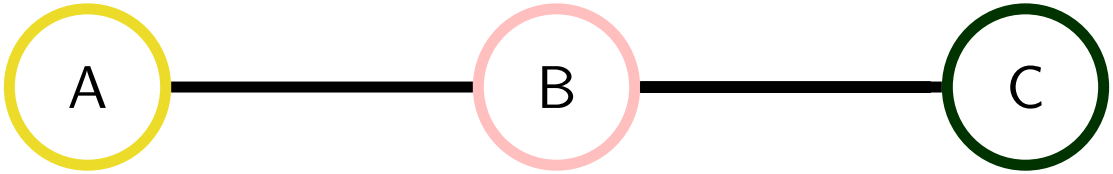
	q_0	q_1	q_2	q_3	q_4	q_5	q_6
Simu	0.29999	0.28283	0.20311	0.11820	0.05791	0.02451	0.00914
m-f	0.30000	0.28471	0.20436	0.11795	0.05693	0.02361	0.00858
Pair	0.30000	0.28288	0.20315	0.11820	0.05788	0.02449	0.00912
Triplet	0.30002	0.28279	0.20312	0.11821	0.05791	0.02451	0.00914

Error "Approx - simu":



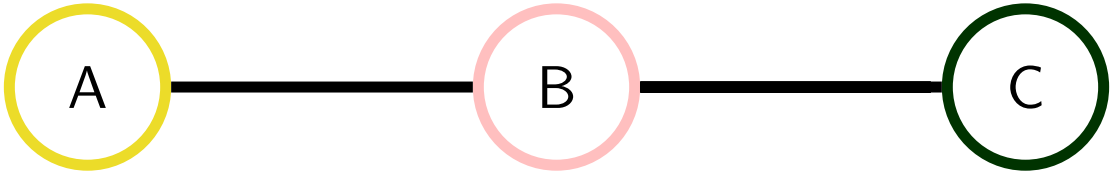
Outline

Refined replica method



We consider N replicas of the same model. When one agent A interacts with B, it interact with one of the N replicas at random.

Refined replica method



We consider N replicas of the same model. When one agent A interacts with B, it interact with one of the N replicas at random.

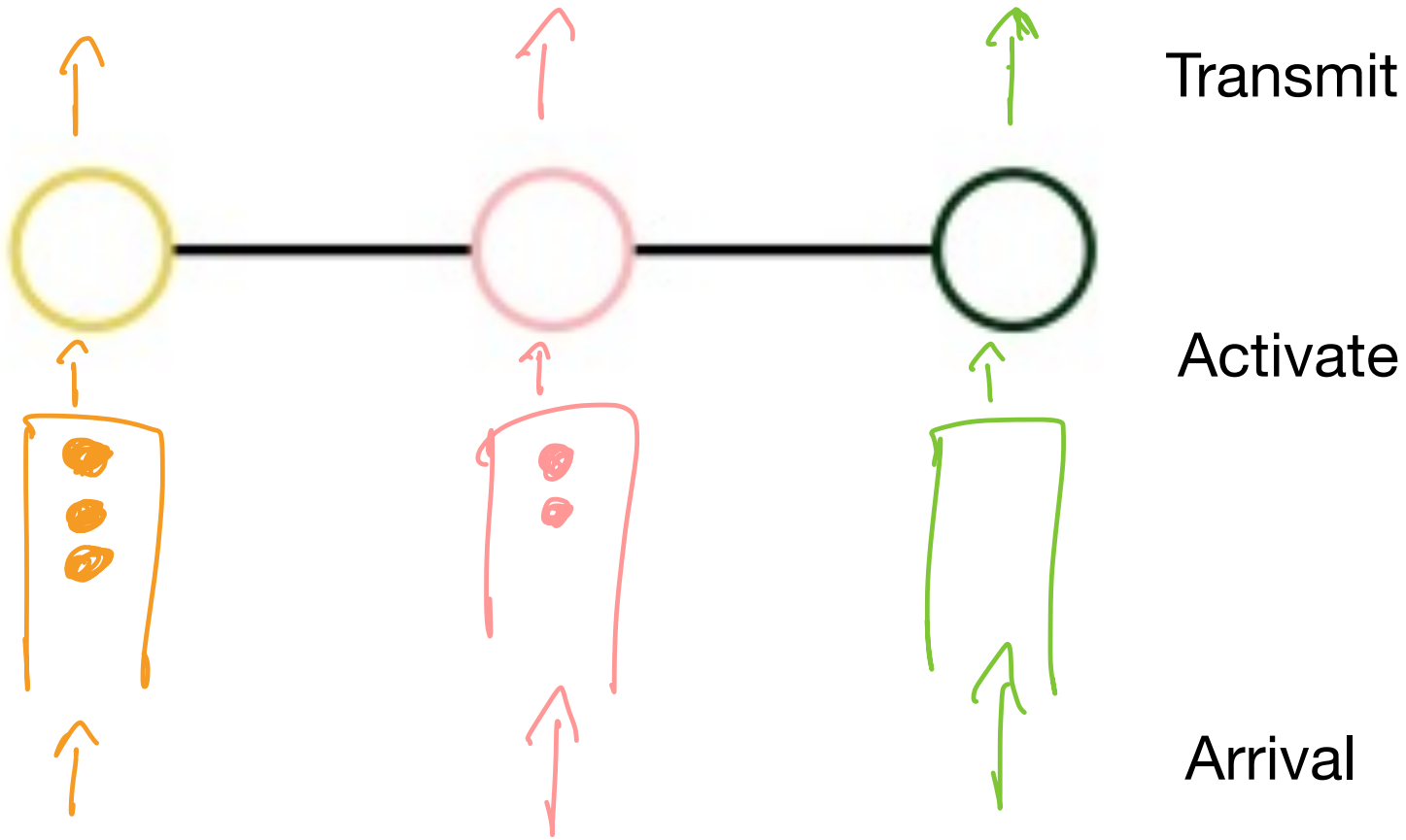
"Theorem": For many systems:

$$\mathbf{P}\left[\text{Agent A is in state } i\right] = \underbrace{\underbrace{x_{A,i}}_{\text{replica mf}} + \frac{1}{N} v_{A,i}}_{\text{refined replica}} + O(1/N^2).$$

This is often very accurate, even for $N = 1$.

Example: two-timescale "replica" mean field

CSMA model from Cecchi et al. 2015

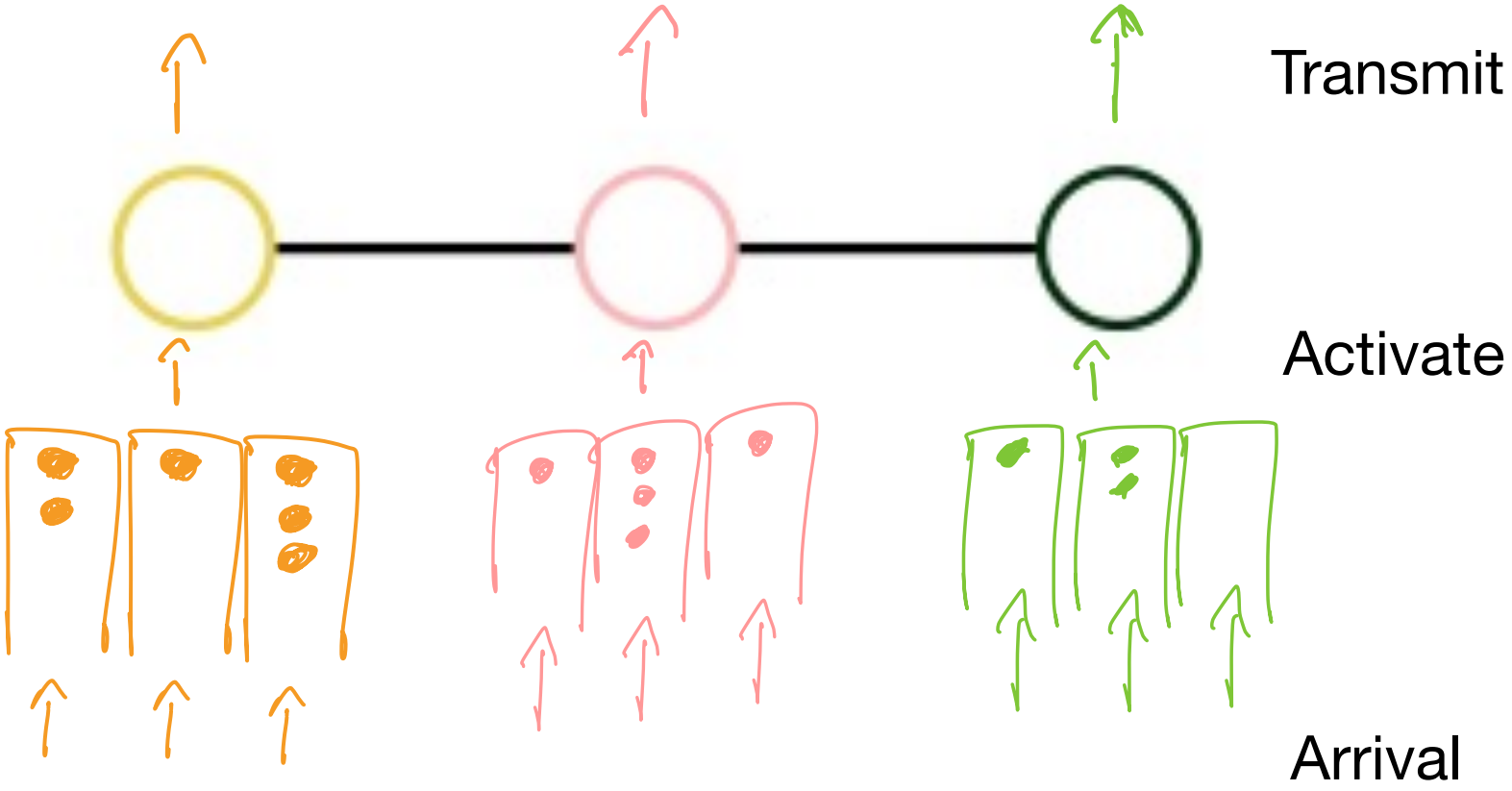


Objective: estimate steady-state.

$$\mathbf{P}\left[S_k = i\right]$$

Example: two-timescale "replica" mean field

CSMA model from Cecchi et al. 2015



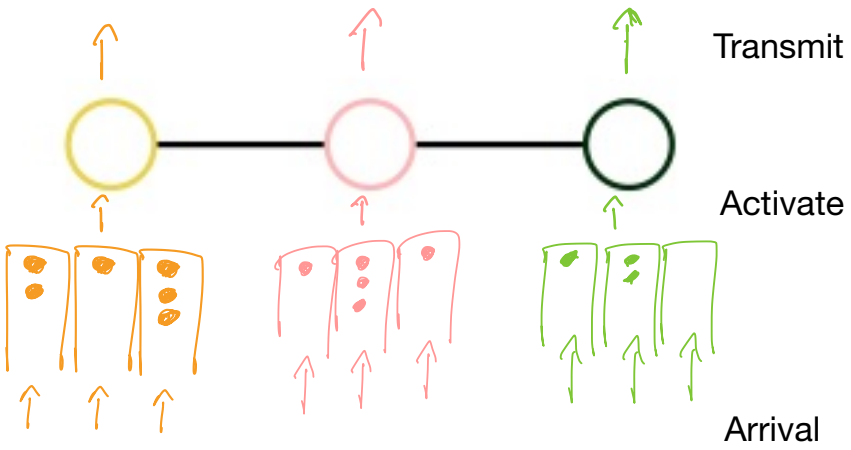
Objective: estimate steady-state.

$$\mathbf{P}\left[S_k = i\right]$$

Scaling: *replica*

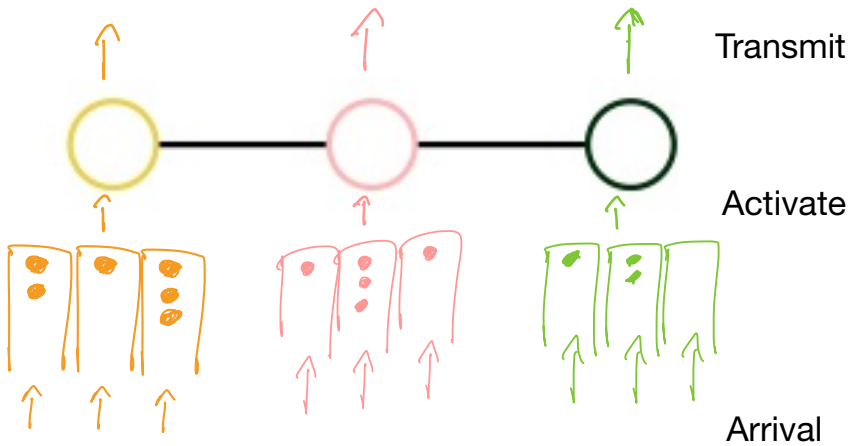
- N severs per node
- Arrival rate $\times \dfrac{1}{N}$.

Illustration of the theorem

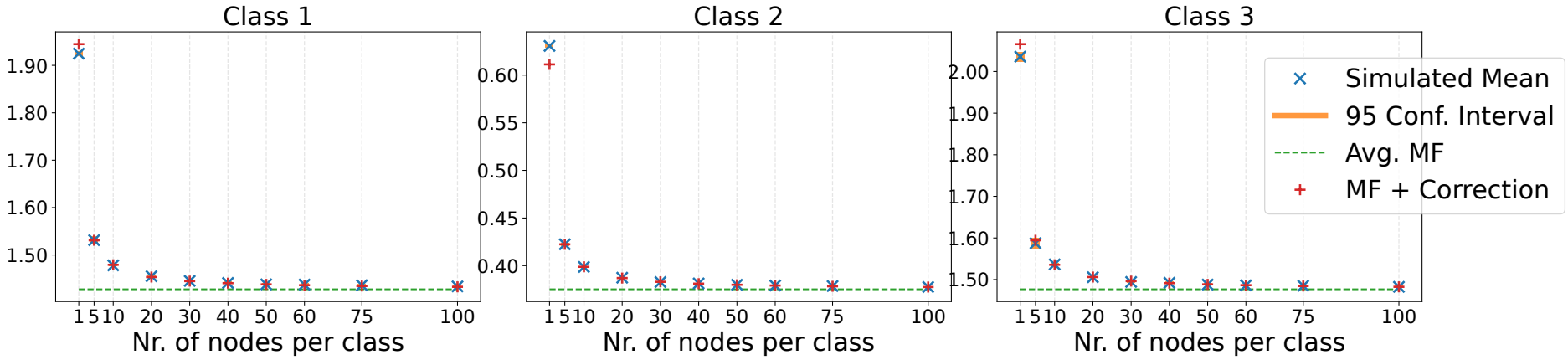


Transmission rates	1.4	1.3	1.7
Activation rates	1.2	2	1.5
Arrival rates	0.5	0.2	0.5

Illustration of the theorem



Transmission rates	1.4	1.3	1.7
Activation rates	1.2	2	1.5
Arrival rates	0.5	0.2	0.5



MF+correction is almost exact

Outline

Two timescale mean field model

Population of N objects.

- Object k has a state $S_k(t) \in \mathcal{S}$.
- Shared resource $Y(t) \in \mathcal{Y}$.

$$X_i = \text{ fraction of objects in state } i.$$

Two timescale mean field model

Population of N objects.

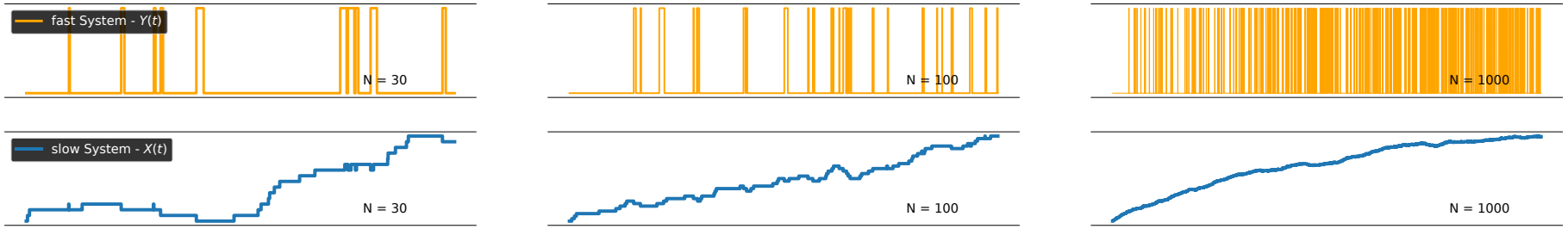
- Object k has a state $S_k(t) \in \mathcal{S}$.
- Shared resource $Y(t) \in \mathcal{Y}$.

$$X_i = \text{ fraction of objects in state } i.$$

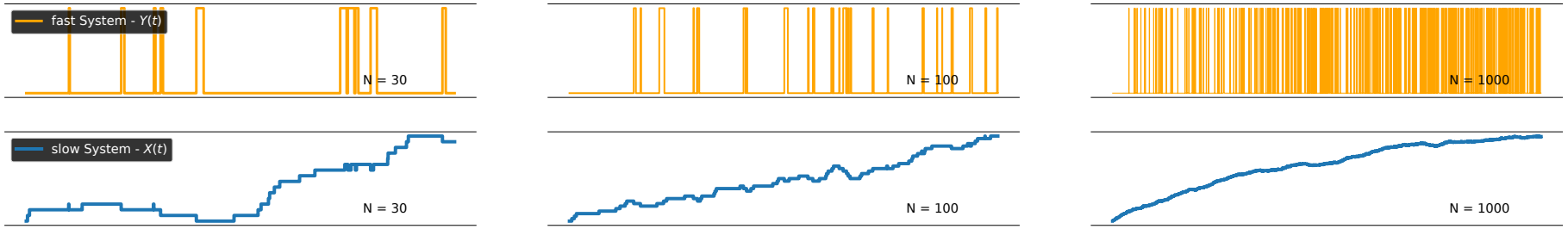
Model

- Object n jumps from i to j at rate $Q_{i,j}(\mathbf{X}, \mathbf{Y})$
- Resource $\mathbf{Y}$ jumps from y to y' at rate $K_{y,y'}(\mathbf{X})$.

We use two approximations to construct a fluid limit.



We use two approximations to construct a fluid limit.



"Average" mean field approximation

Let $\pi_y(\mathbf{x})$ be the stationary distribution of $K(\mathbf{x})$. We define:

$$\bar{Q}(\mathbf{x}) = \sum_y \pi_y(\mathbf{x}) Q(\mathbf{x}, y).$$

The mean field approximation is the solution of the ODE:

$$\dot{\mathbf{x}} = \mathbf{x}\bar{Q}(\mathbf{x}),$$

The approximation is asymptocally exact. Its bias is v/N.

Assume that

- K is unichain for all $\mathbf{x}$.
- K and Q are twice differenciable
- $\dot{\mathbf{x}} = \mathbf{x}\bar{Q}(\mathbf{x})$ has a unique attractor x^*.

Theorem

There exists a *computable* V such that, in steady-state:

$$\mathbf{P}\left[S_k = i\right] = \underbrace{\underbrace{x_i^*}_{\text{mean field approximation}} + \frac{1}{n}V_i}_{\text{refined approximation}} + O(1/n^2).$$

If Y is not here, we can directly use Stein's method

Let G^{sto} be the Generator of the stochastic system. For $h : \mathcal{X} \to \mathbb{R}$:

1. $$G^{\mathrm{sto}} h(X) = \sum_{i,j} (h(X + \frac{1}{n}(e_j - e_i)) - h(X)) n x_i Q_{ij}(x)$$

If Y is not here, we can directly use Stein's method

Let G^{sto} be the Generator of the stochastic system. For $h : \mathcal{X} \to \mathbb{R}$:

1. $$\begin{aligned} G^{\mathrm{sto}} h(X) &= \sum_{i,j} (h(X + \frac{1}{n}(e_j - e_i)) - h(X)) n x_i Q_{ij}(x) \\ &= \underbrace{\nabla h \cdot x Q(X)}_{\text{Generator of ODE } \dot{x} = xQ(x).} + \underbrace{O(1/n)}_{\text{if } h \text{ is } C^1}. \end{aligned}$$

2. $\mathbb{E}\left[G^{\mathrm{sto}} h(X)\right] = 0$ if X is in steady-state.

If Y is not here, we can directly use Stein's method

and introduce a Poisson equation for the slow system

Let G^{sto} be the Generator of the stochastic system. For $h : \mathcal{X} \to \mathbb{R}$:

1. $$G^{\mathrm{sto}} h(X) = \sum_{i,j} (h(X + \frac{1}{n}(e_j - e_i)) - h(X)) n x_i Q_{ij}(x)$$
$$= \underbrace{\nabla h \cdot xQ(X)}_{\text{Generator of ODE } \dot{x} = xQ(x).} + \underbrace{O(1/n)}_{\text{if } h \text{ is } C^1}.$$

2. $\mathbb{E}\left[G^{\mathrm{sto}} h(X)\right] = 0$ if X is in steady-state.

Let G be such that $\nabla G \cdot xQ(x) = x - x^*$ (Poisson equation). We have:

$$\begin{aligned}
\mathbb{E}\left[X - x^*\right] &= \mathbb{E}\left[\nabla G \cdot XQ(X)\right] \\
&= \mathbb{E}\left[(\nabla G - G^{\mathrm{sto}}) \cdot XQ(X)\right] && \text{(by (2))} \\
&= O(1/n) && \text{(by (1))}.
\end{aligned}$$

When Y is here, we need to treat the fast system

Let $h : \mathcal{X} \to \mathbb{R}$ be a test function. We have:

$$\begin{aligned} G^{\mathrm{sto}} h(X, Y) &= \sum_{i,j} (h(X + \frac{1}{n}(e_j - e_i)) - h(X)) n x_i Q_{ij}(X, Y) \\ &= \underbrace{\nabla h \cdot XQ(X, Y)}_{\neq G^{\mathrm{ODE}} = \nabla h \cdot \bar{Q}(x).} + O(1/n), \end{aligned}$$

When Y is here, we need to treat the fast system

Let $h : \mathcal{X} \to \mathbb{R}$ be a test function. We have:

$$\begin{aligned} G^{\mathrm{sto}} h(X, Y) &= \sum_{i,j} (h(X + \frac{1}{n}(e_j - e_i)) - h(X)) n x_i Q_{ij}(X, Y) \\ &= \underbrace{\nabla h \cdot X Q(X, Y)}_{\neq G^{\mathrm{ODE}} = \nabla h \cdot \bar{Q}(x).} + O(1/n), \end{aligned}$$

We are left with $Q(X, Y) - \bar{Q}(X)$.

When Y is here, we need to treat the fast system

and introduce a Poisson equation for the fast system.

Let $h : \mathcal{X} \to \mathbb{R}$ be a test function. We have:

$$G^{\mathrm{sto}} h(X,Y) = \sum_{i,j} (h(X + \frac{1}{n}(e_j - e_i)) - h(X)) n x_i Q_{ij}(X,Y)$$
$$= \underbrace{\nabla h \cdot X Q(X,Y)}_{\neq G^{\mathrm{ODE}} = \nabla h \cdot \bar{Q}(x).} + O(1/n),$$

We are left with $Q(X,Y) - \bar{Q}(X)$.

Lemma: There exists a K^+ that is C^2 such that for all $h : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$:

$$h(X,Y) - \bar{h}(X) = K(x) K^+(x) h(X,Y).$$

Rapping up the proof

Let $h : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ be a test function.

$$G^{\mathrm{sto}} h(X, Y) = \underbrace{nK(X)h(X, Y)}_{\text{fast}} + \underbrace{\nabla_x h \cdot XQ(X, Y)}_{\text{slow}} + O(1/n)$$

Hence, $K^{fast} - \frac{1}{n} G^{\mathrm{sto}} = o(1/n)$ if h is C^1.

Rapping up the proof

Let $h : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ be a test function.

$$G^{\mathrm{sto}} h(X, Y) = \underbrace{nK(X)h(X, Y)}_{\text{fast}} + \underbrace{\nabla_x h \cdot XQ(X, Y)}_{\text{slow}} + O(1/n)$$

Hence, $K^{fast} - \frac{1}{n} G^{\mathrm{sto}} = o(1/n)$ if h is C^1.

This shows that in steady-state:

$$\begin{aligned}
\mathbb{E}\left[h(X) - \bar{h}(X)\right] &= \mathbb{E}\left[K(x)K^+(x)h(X, Y)\right] \\
&= \mathbb{E}\left[(K(x) - \frac{1}{n} G^{\mathrm{sto}})K^+(x)h(X, Y)\right] && \text{(steady-state)}. \\
&= O(1/n) && \text{(expansion above)}.
\end{aligned}$$

Outline

Conclusion

Interacting models on graphs are complicated.
We studied two heuristic methods:

- The pair approximation.
- The "refined replica" method.

Conclusion

Interacting models on graphs are complicated.
We studied two heuristic methods:

- The pair approximation.
- The "refined replica" method.

For both cases:

- No real guarantee of why it is so accurate.
- Help wanted!

Slides and references: `http://polaris.imag.fr/nicolas.gast`

References

Results on which the second part of this talk is based:

- Bias and Refinement of Multiscale Mean Field Models. Allmeier, Gast, 2022. Sigmetrics 2023.
  - With a model from CSMA networks in a many-sources regime: A mean-field approach. Cecchi, Borst, van Leeuwaarden, Whiting. Infocom 2016.

(for first part, see upcoming preprint on arXiv).

Related refined mean-field approximation papers:

- Mean Field and Refined Mean Field Approximations for Heterogeneous Systems: It Works! by Allmeier and Gast. SIGMETRICS 2022.
- A Refined Mean Field Approximation by Gast and Van Houdt. SIGMETRICS 2018.