A Tutorial on Mean Field and Refined Mean Field Approximation

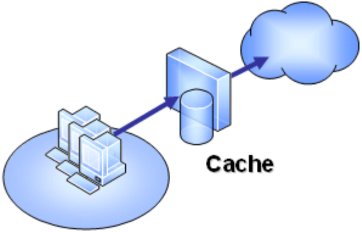
Nicolas Gast

Inria, Grenoble, France

ITC 31, August 2019, Budapest

Good system design needs performance evaluation

Example : caches



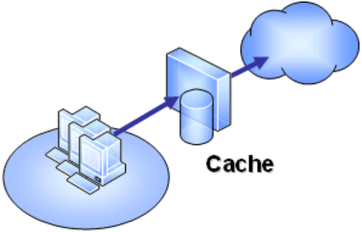
N contents

Which cache replacement policies?

- LRU
- 2-LRU
- RAND
- RAND(m)
- FIFO

Good system design needs performance evaluation

Example : caches



N contents

Which cache replacement policies?

- LRU
- 2-LRU
- RAND
- RAND(m)
- FIFO

We need models and methods to characterize emerging behavior starting from a stochastic model of interacting objects

- We use ~~simulation~~ analytical methods and approximations.

The main difficulty of probability : correlations

$$\mathbf{P}[A, B] \neq \mathbf{P}[A]\,\mathbf{P}[B]$$

Problem: state space explosion
S states per object, N objects $\Rightarrow S^N$ states

"Mean field approximation" $\approx$ assume that all objects are independent

Where has it been used?

- Statistical mechanics, chemical reaction networks Gillepsie 92
- Communication protocols ex: CSMA, Bianchi 00
- Performance of caching algorithms ex: TTL-approximation, Fagins 77 + many recent papers
- Mean field games ex: evacuation, Mexican wave, Larsi-Lions 06
- Understanding of Deep Neural Networks ex: Xiao et al. 2018

"Mean field approximation" $\approx$ assume that all objects are independent

Where has it been used?

- Statistical mechanics, chemical reaction networks Gillepsie 92
- Communication protocols ex: CSMA, Bianchi 00
- Performance of caching algorithms ex: TTL-approximation, Fagins 77 + many recent papers
- Mean field games ex: evacuation, Mexican wave, Larsi-Lions 06
- Understanding of Deep Neural Networks ex: Xiao et al. 2018

This talk

1. What is mean field approximation and how to apply it?
2. When is it valid?
3. How can we do better?

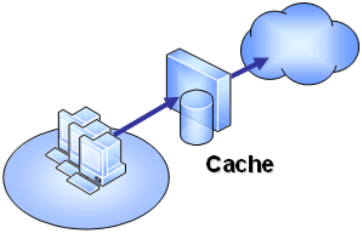
Outline: Demystifying Mean Field Approximation

1. How to Construct a Mean Field Approximation (via an example)
2. How Accurate is the Classical Mean Field Approximation
3. We can Refine this Approximation
4. Conclusion and Open Questions

Outline

1. How to Construct a Mean Field Approximation (via an example)
   - Approach 1: Zoom on One Object and Independence
   - Approach 2: Density Dependent Population Processes
   - Approach 3: Discrete-Time Models
2. How Accurate is the Classical Mean Field Approximation
   - Transient Analysis
   - Steady-state Regime
3. We can Refine this Approximation
   - Main Results
   - Idea of the Proof: System Size Expansion
4. Conclusion and Open Questions

Analysis of Cache (Re)placement Policies

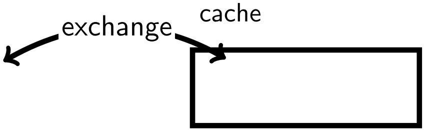


- Popularity-oblivious policies (LRU, RANDOM[1], TTL-caches[2])
- Popularity-aware policies / learning (LFU and variants[3], network of caches[4])

[1] started with [King 1971, Gelenbe 1973]
[2] *e.g.*, Fofack et al 2013, Berger et al. 2014
[3] Optimizing TTL Caches under Heavy-Tailed Demands (Ferragut et al. SIGMETRICS 2016)
[4] Adaptive Caching Networks with Optimality Guarantees (Ioannidis and Yeh, SIGMETRICS16)

The RAND($\vec{m}$) policy [5]

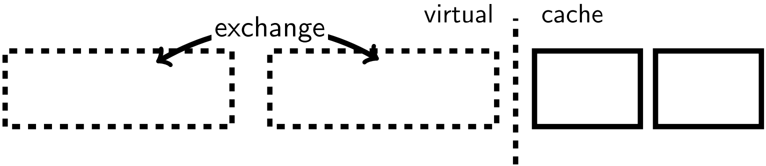
- RANDOM: exchange the requested item with a random item



[5] Can be generalized in LRU($\vec{m}$) or FIFO($\vec{m}$), see [ITC16,QUESTA16]

The RAND($\vec{m}$) policy [5]

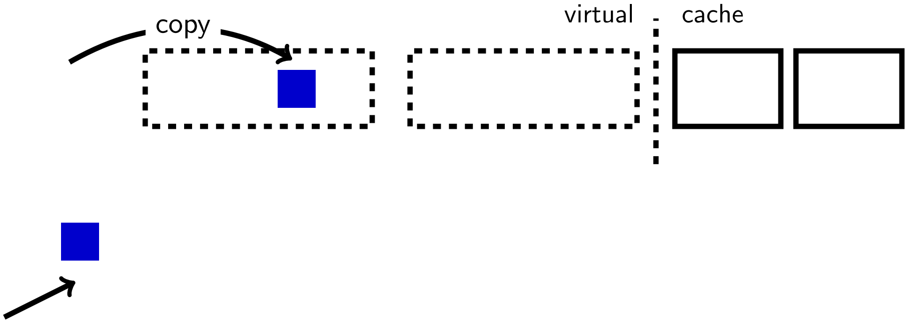
- RAND($\vec{m}$): exchange the requested item with an item from next list



[5] Can be generalized in LRU($\vec{m}$) or FIFO($\vec{m}$), see [ITC16,QUESTA16]

The RAND($\vec{m}$) policy [5]

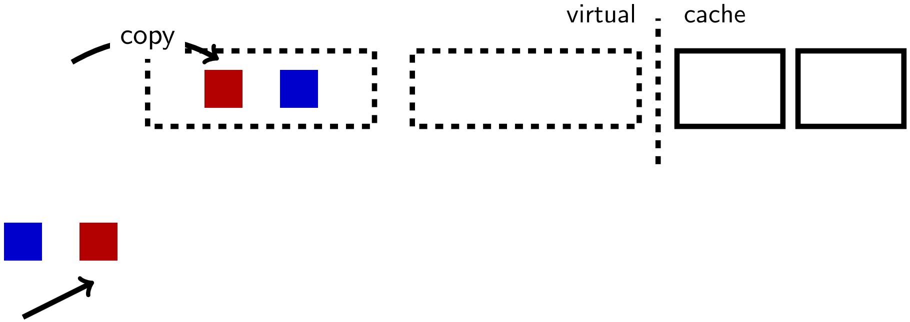
- RAND($\vec{m}$): exchange the requested item with an item from next list



[5] Can be generalized in LRU($\vec{m}$) or FIFO($\vec{m}$), see [ITC16,QUESTA16]

The RAND($\vec{m}$) policy [5]

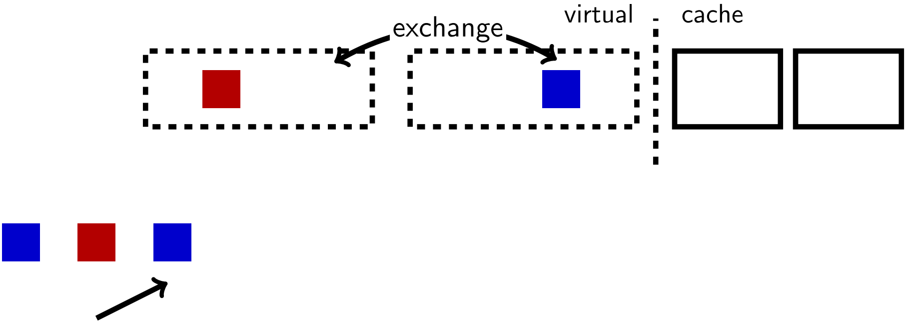
- RAND($\vec{m}$): exchange the requested item with an item from next list



[5] Can be generalized in LRU($\vec{m}$) or FIFO($\vec{m}$), see [ITC16,QUESTA16]

The RAND($\vec{m}$) policy [5]

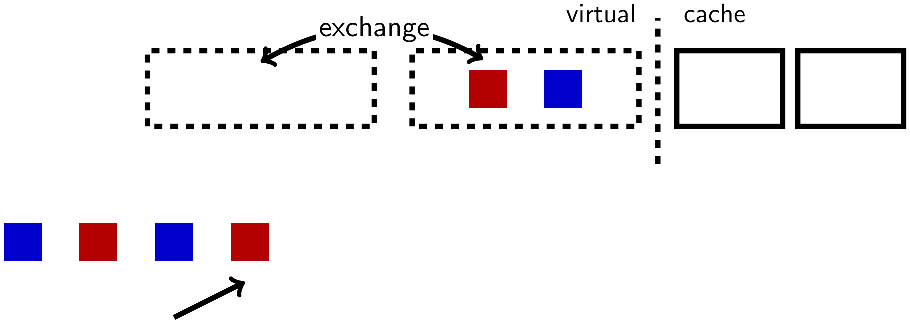
- RAND($\vec{m}$): exchange the requested item with an item from next list



[5] Can be generalized in LRU($\vec{m}$) or FIFO($\vec{m}$), see [ITC16,QUESTA16]

The RAND($\vec{m}$) policy [5]

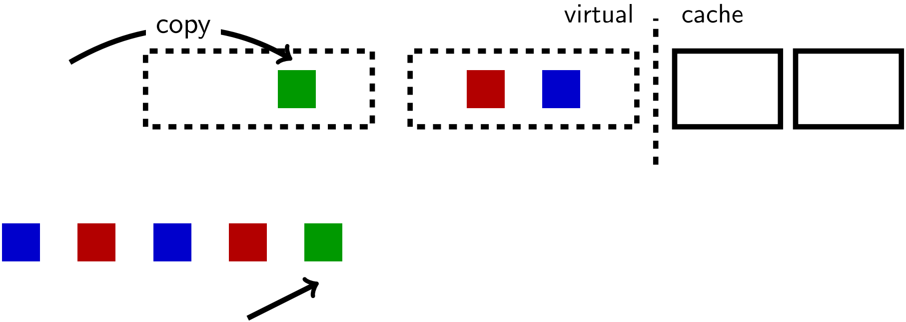
- RAND($\vec{m}$): exchange the requested item with an item from next list



[5] Can be generalized in LRU($\vec{m}$) or FIFO($\vec{m}$), see [ITC16,QUESTA16]

The RAND($\vec{m}$) policy [5]

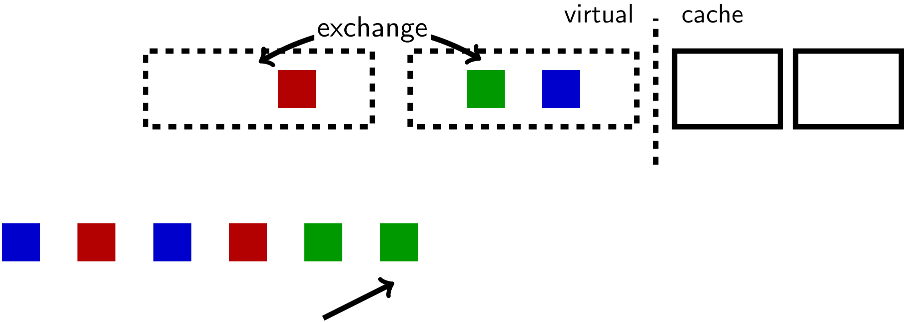
- RAND($\vec{m}$): exchange the requested item with an item from next list



[5] Can be generalized in LRU($\vec{m}$) or FIFO($\vec{m}$), see [ITC16,QUESTA16]

The RAND($\vec{m}$) policy [5]

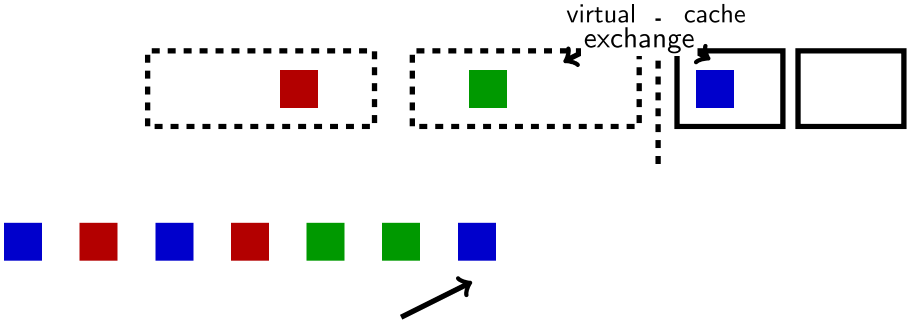
- RAND($\vec{m}$): exchange the requested item with an item from next list



[5] Can be generalized in LRU($\vec{m}$) or FIFO($\vec{m}$), see [ITC16,QUESTA16]

The RAND($\vec{m}$) policy [5]

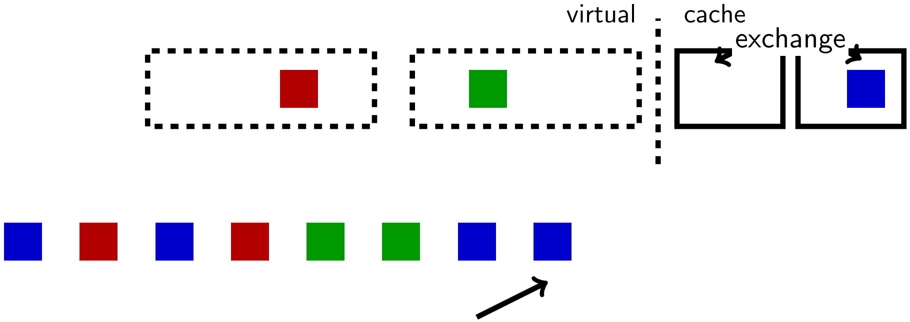
- RAND($\vec{m}$): exchange the requested item with an item from next list



[5] Can be generalized in LRU($\vec{m}$) or FIFO($\vec{m}$), see [ITC16,QUESTA16]

The RAND($\vec{m}$) policy [5]

- RAND($\vec{m}$): exchange the requested item with an item from next list



[5] Can be generalized in LRU($\vec{m}$) or FIFO($\vec{m}$), see [ITC16,QUESTA16]

Our model is a heterogeneous population model

N items of identical sizes but different popularities.

- The state of an item is the list in which it is $\{\emptyset\} \cup \{1, \ldots, h\}$.
- The item k is requested at rate λ_k.

Our model is a heterogeneous population model

N items of identical sizes but different popularities.

- The state of an item is the list in which it is $\{\emptyset\} \cup \{1, \dots, h\}$.
- The item k is requested at rate λ_k.

We write: $X_{k,i}(t) = 1$ if object k is in list i.

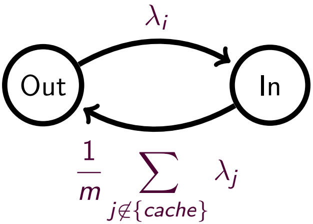
We want to find an approximation of $\mathbb{E}\left[X_{k,i}(t)\right]$

- ($=$ probability that the object k is in state i).

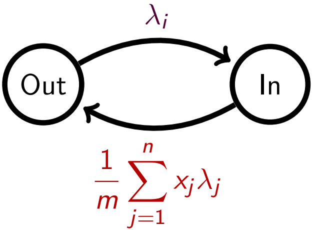
Construction of the mean-field approximation (RANDOM)

$$\begin{tikzcd} \text{Out} \arrow[r, bend left, "\lambda_i"] & \text{In} \arrow[l, bend left, "\frac{1}{m}\sum_{j\notin\{cache\}}\lambda_j"] \end{tikzcd}$$

Construction of the mean-field approximation (RANDOM)

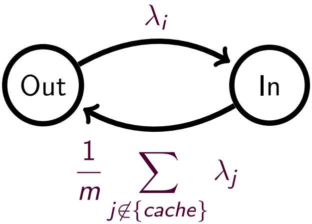


Original model for RANDOM
(only one list)

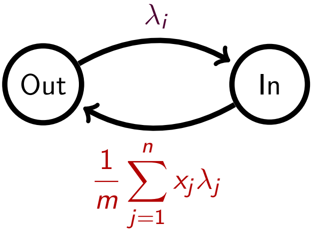


MF approx: let $x_i(t) = \mathbf{P}\left[i \notin \{cache\}\right]$.
If all objects are independent:

Construction of the mean-field approximation (RANDOM)



Original model for RANDOM
(only one list)



MF approx: let $x_i(t) = \mathbf{P}\left[i \notin \{cache\}\right]$.
If all objects are independent:

The "mean field" equations for the approximation model are:

$$\dot{x}_i = -\lambda_i x_i + \frac{1}{m} \sum_{j=1}^{n} x_j(t) \lambda_j (1 - x_i).$$

Steady-state analysis for RANDOM

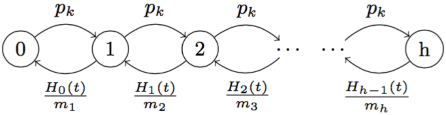
The mean field equation $\dot{x}_i = -\lambda_i x_i + \frac{1}{m}\sum_{j=1}^{n} x_j(t)\lambda_j(1-x_i)$. has a unique fixed point that satisfies:

$$\pi_i = \frac{z}{z+\lambda_i} \qquad \text{with } z \text{ such that} \sum_{i=1}^{n}(1-\pi_i) = m.$$

Same equations as Fagins (77).

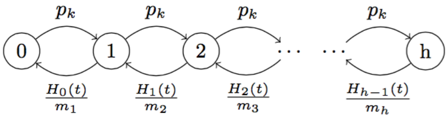
Extension to the RAND(m) model (G, Van Houdt SIGMETRICS 2015)

Let $H_i(t)$ be the popularity in list i.



Extension to the RAND(m) model (G, Van Houdt SIGMETRICS 2015)

Let $H_i(t)$ be the popularity in list i.



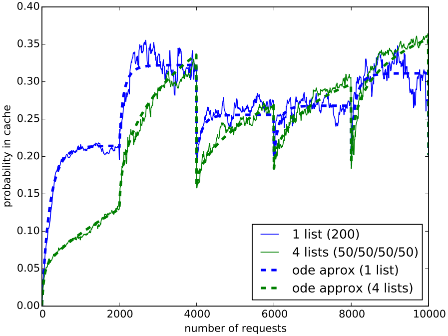
If $x_{k,i}(t)$ is the probability that item k is in list i at time t, we approximately have:

$$\dot{x}_{k,i}(t) = p_k x_{k,i-1}(t) - \overbrace{\boxed{\sum_j p_j x_{j,i-1}(t)}}^{\text{Popularity in cache i-1}} \frac{x_{k,i}(t)}{m_i} + \mathbf{1}_{\{i<h\}} \Bigg(\underbrace{\boxed{\sum_j p_j x_{j,i}(t)}}_{\text{Popularity in cache i}} \frac{x_{k,i+1}(t)}{m_{i+1}} - p_k x_{k,i}(t) \Bigg)$$

This approximation is of the form $\dot{x} = xQ(x)$.

The mean field approximation is very accurate

$n = 1000$ objects with Zipf popularities.



The popularities change every 2000 requests

m_1	m_2	m_3	m_4	exact	mean field
2	2	96	–	0.3166	0.3169
10	30	60	–	0.3296	0.3299
20	2	78	–	0.3273	0.3276
90	8	2	–	0.4094	0.4100
1	4	10	85	0.3039	0.3041
5	15	25	55	0.3136	0.3139
25	25	25	25	0.3345	0.3348
60	2	2	36	0.3514	0.3517

Steady-state miss probabilities

A population model

Consider now that we have K types of items with N items per type.

- The state of an item is the list in which it is $\{\emptyset\} \cup \{1, \dots, h\}$.
- An item of type k is requested at rate λ_k.

A population model

Consider now that we have K types of items with N items per type.

- The state of an item is the list in which it is $\{\emptyset\} \cup \{1, \dots, h\}$.
- An item of type k is requested at rate λ_k.

We define $X_{k,i}(t)$ has the fraction of type k items in list i.

The transitions are:

$$\mathbf{X} \mapsto \mathbf{X} + \frac{1}{N}\left(-e_{k,i} + e_{k,i+1} + e_{\ell,i} - e_{\ell,i+1}\right) \text{ at rate } N\lambda_k X_{k,i} \frac{X_{\ell,i+1}}{m_{i+1}}$$

This is an example of a density dependent population process (Kurtz, 70s)

A population process is a sequence of CTMCs $X^N(t)$ indexed by the population size N, with state space $E^N \subset E$ and transitions (for $\ell \in \mathcal{L}$):

$$X \mapsto X + \frac{\ell}{N} \qquad \text{at rate } N\beta_\ell(X).$$

This is an example of a density dependent population process (Kurtz, 70s)

A population process is a sequence of CTMCs $X^N(t)$ indexed by the population size N, with state space $E^N \subset E$ and transitions (for $\ell \in \mathcal{L}$):

$$X \mapsto X + \frac{\ell}{N} \qquad \text{at rate } N\beta_\ell(X).$$

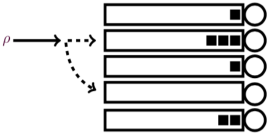
The Mean field approximation

The drift is $f(x) = \sum_\ell \ell\beta_\ell(x)$ and the mean field approximation is the solution of the ODE $\dot{x} = f(x)$.

Other examples of density dependent population processes

Load Balancing and JSQ(2)

$$X_i(t) = \frac{1}{N}\sum_{n=1}^{N} \mathbf{1}_{\{S_n(t)\geq i\}} = \text{fraction of queues with queue length} \geq i.$$

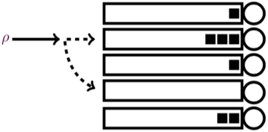


- Arrival: $x \mapsto x + \frac{1}{N}\mathbf{e_i}$.
- Departures: $x \mapsto x - \frac{1}{N}\mathbf{e_i}$.

Other examples of density dependent population processes

Load Balancing and JSQ(2)

$$X_i(t) = \frac{1}{N}\sum_{n=1}^{N} \mathbf{1}_{\{S_n(t)\geq i\}} = \text{fraction of queues with queue length} \geq i.$$



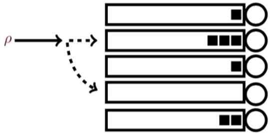
- Arrival: $x \mapsto x + \frac{1}{N}\mathbf{e_i}$.
- Departures: $x \mapsto x - \frac{1}{N}\mathbf{e_i}$.

Pick two servers at random, what is the probability the least loaded has $i-1$ jobs?

Other examples of density dependent population processes

Load Balancing and JSQ(2)

$$X_i(t) = \frac{1}{N}\sum_{n=1}^{N} \mathbf{1}_{\{S_n(t)\geq i\}} = \text{fraction of queues with queue length } \geq i.$$



- Arrival: $x \mapsto x + \frac{1}{N}\mathbf{e_i}$.
- Departures: $x \mapsto x - \frac{1}{N}\mathbf{e_i}$.

Pick two servers at random, what is the probability the least loaded has $i-1$ jobs?

$x_{i-1}^2 - x_i^2$ when picked with replacement

$x_{i-1}\frac{Nx_{i-1}-1}{N-1} - x_i\frac{Nx_i-1}{N-1}$ when picked without replacement

Note: this becomes asymptotically the same as N goes to infinity.

Transitions and Mean Field Approximation

State changes on x:

$$x \mapsto x + \frac{1}{N}\mathbf{e_i} \text{ at rate } N\rho(x_{i-1}^2 - x_i^2)$$
$$x \mapsto x - \frac{1}{N}\mathbf{e_i} \text{ at rate } N(x_i - x_{i+1})$$

The mean field approximation is to consider the ODE associated with the drift (average variation):

$$\dot{x}_i = \underbrace{\rho(x_{i-1}^2 - x_i^2)}_{\text{Arrival}} - \underbrace{(x_i - x_{i+1})}_{\text{Departure}}$$

There is a unique fixed point (which also is an attractor):

$$\pi_i = \rho^{2^i - 1}$$

Outline

Benaïm-Le Boudec's model (PEVA 2007)

Time is discrete.

$$X_i(k) = \text{Proportion of object in state } i \text{ at time step } k$$

$$R(k) = \text{State of the "resource" at time } k \text{ (discrete)}$$

Benaïm-Le Boudec's model (PEVA 2007)

Time is discrete.

$$X_i(k) = \text{Proportion of object in state } i \text{ at time step } k$$
$$R(k) = \text{State of the "resource" at time } k \text{ (discrete)}$$

Assumptions:

- Only $O(1)$ objects change state at each time step and

$$f(x,r) = \frac{1}{N}\mathbb{E}\left[X(k+1) - X(k) | X(k) = x, R(k) = r\right].$$

- R evolves fast in a discrete state-space and:

$$\mathbf{P}\left[R(k+1) = j | X(k) = x, R(k) = i\right] = P_{ij}(x).$$

For all x, $P(x)$ is irreducible and has a unique stationary measure $\pi(x,.)$.

Mean Field Approximation

Examples with resource: CSMA protocols, Opportunistic networks.

$$\dot{x} = \sum_r f(x, r)\pi(x, r),$$

where $\pi(x, r)$ is the stationary measure of the resource given x.

Mean Field Approximation

Examples with resource: CSMA protocols, Opportunistic networks.

$$\dot{x} = \sum_r f(x,r)\pi(x,r),$$

where $\pi(x,r)$ is the stationary measure of the resource given x.

The analysis of such models is done by considering stochastic approximation algorithms. For example, without resource one has:

$$X(k+1) = X(k) + \frac{1}{N}\left[f(X(k)) + M(k+1)\right],$$

where M is some noise process.
This is a noisy Euler discretization of an ordinary differential equation.

Take-home message on this part

Three ways to construct mean field approximation:

- Independence assumption $\dot{x} = xQ(x)$.
- Density dependent population process.
- Discrete-time model with vanishing intensity.

In what follows, I will assume that X is a density dependent population process (ex: $SQ(d)$). Analysis of other models are similar.

Outline

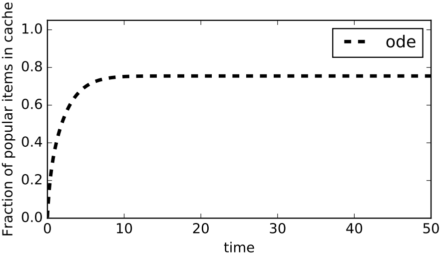
Convergence Result as N Goes to Infinity

Theorem (under technical conditions), for any density dependent population process X^N whose drift is Lipschitz-continuous, if $X^N(0)$ converges to x_0, then for any finite T:

$$\sup_{0 \leq t \leq T} \left\| X^N(t) - x(t) \right\| \to 0.$$

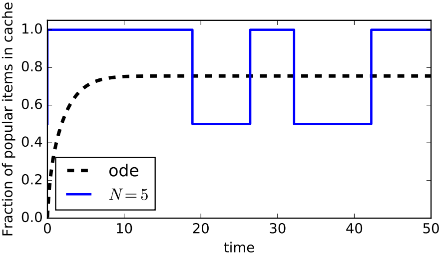
where $x(t)$ is the unique solution of the ODE $\dot{x} = f(x)$.

Illustration (cache model)



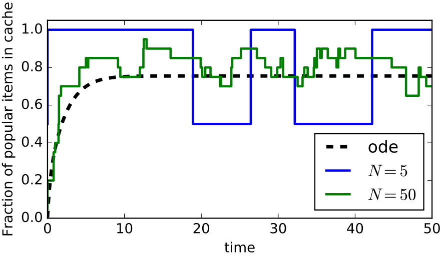
Example. Fraction of popular items in cache. $m_1 = N/4$, $m_2 = N/4$, $N/2$ popular object (popularity $4N/(5N)$, $N/2$ non-popular objects (popularity $N/(5N)$).

Illustration (cache model)



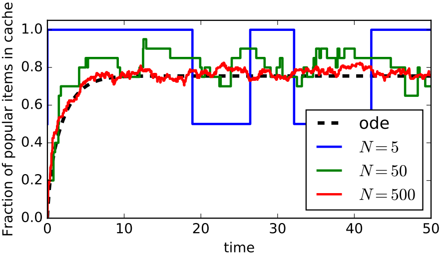
Example. Fraction of popular items in cache. $m_1 = N/4$, $m_2 = N/4$, $N/2$ popular object (popularity $4N/(5N)$, $N/2$ non-popular objects (popularity $N/(5N)$).

Illustration (cache model)



Example. Fraction of popular items in cache. $m_1 = N/4$, $m_2 = N/4$, $N/2$ popular object (popularity $4N/(5N)$, $N/2$ non-popular objects (popularity $N/(5N)$).

Illustration (cache model)



Example. Fraction of popular items in cache. $m_1 = N/4$, $m_2 = N/4$, $N/2$ popular object (popularity $4N/(5N)$, $N/2$ non-popular objects (popularity $N/(5N)$).

The fixed point method

Markov chain

Transient regime $\dot{p} = pK$

$$\Big\downarrow\ t \to \infty$$

Stationary $\pi K = 0$

The fixed point method

	Markov chain		Mean-field
Transient regime	$\dot{p} = pK$	$\xrightarrow{N \to \infty}$	$\dot{x} = xQ(x)$
	$\downarrow t \to \infty$		
Stationary	$\pi K = 0$	$\xrightarrow{?}$	$\boxed{x^* Q(x^*) = 0}$ fixed points

Method was used in many papers:

- Bianchi 00, Performance analysis of the IEEE 802.11 distributed coordination function.
- Ramaiyan et al. 08, Fixed point analys is of single cell IEEE 802.11e WLANs: Uniqueness, multistability.
- Kwak et al. 05, Performance analysis of exponenetial backoff.
- Kumar et al 08, New insights from a fixed-point analysis of single cell IEEE 802.11 WLANs.

The fixed point method

$$\begin{tikzcd}
 & \text{Markov chain} & \text{Mean-field} \\
\text{Transient regime} & \dot{p} = pK \arrow[r, "N \to \infty"] \arrow[d, "t \to \infty"] & \dot{x} = xQ(x) \arrow[d, "\cancel{t \to \infty}"] \\
\text{Stationary} & \pi K = 0 \arrow[r, "\cancel{N \to \infty}"] & \boxed{\cancel{x^* Q(x^*) = 0}}
\end{tikzcd}$$

fixed points

The fixed point method

$$
\begin{tikzcd}
 & \text{Markov chain} & \text{Mean-field} \\
\text{Transient regime} & \dot{p} = pK \arrow[r, "N \to \infty"] \arrow[d, "t \to \infty"] & \dot{x} = xQ(x) \arrow[d, "\cancel{t \to \infty}"] \arrow[d, bend left, "\text{if yes}"] \\
\text{Stationary} & \pi K = 0 \arrow[r, "\cancel{N \to \infty}"] \arrow[r, bend right, "\text{then yes}"'] & \underbrace{\boxed{x^* Q(x^*) = 0}}_{\text{fixed points}}
\end{tikzcd}
$$

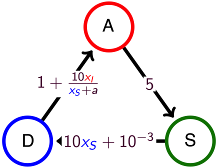
Theorem (Benaim Le Boudec 08)

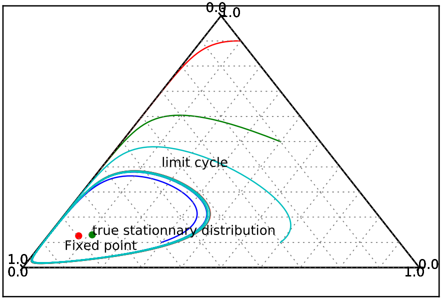
If all trajectories of the ODE converges to the fixed points, the stationary distribution π^N concentrates on the fixed points

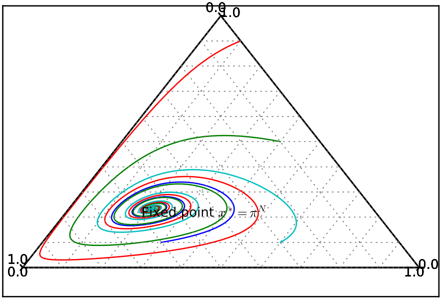
In that case, we also have:

$$\lim_{N\to\infty} \mathbf{P}\left[S_1 = i_1 \ldots S_k = i_k\right] = x_1^* \ldots x_k^*.$$

The fixed point method does not always work







The fixed point methods does not work for $a = 0.1$

It works for $a = 0.3$

Counter-examples for specific CSMA models (Cho, Le Boudec, Jiang 2011)

Fixed point method in practice

From the examples coming from queuing theory, many models have a unique attractor.

- This holds for classical load balancing policies such as SQ(d), pull-push, JIQ,...
  - Often comes from monotonicity
- This holds in many cases in statistical physics
  - Lyapunov methods (entropy, reversibility) (ex: G. 2016 for cache)
- It does not always work
  - Theoretical biology / chemistry
  - Multi-stable models (ex: Kelly)
  - Counter-examples for specific CSMA models (Cho, Le Boudec, Jiang 2011)

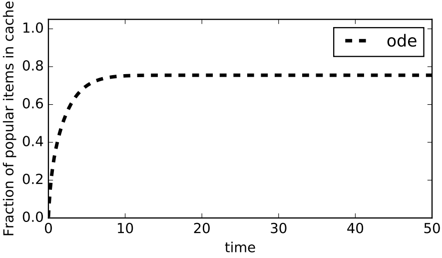
Outline

For which system size N can mean-field be used?

Recall: $X_i^N(t)$ = fraction of objects in state i.

Theorem (Kurtz (1970s), Ying (2016)):

If the drift f is Lipschitz-continuous:	If in addition the ODE has a unique attractor π:
$$X^N(t) \approx x(t) + \frac{1}{\sqrt{N}} G_t$$	$$\mathbb{E}\left[X^N(\infty) - \pi\right] = O(1/\sqrt{N})$$



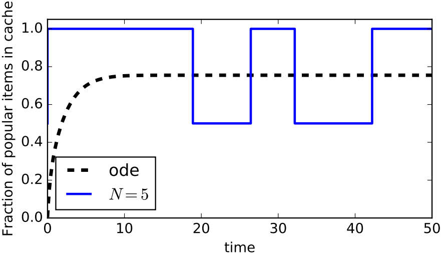
Example. Fraction of popular items in cache. $m_1 = N/4$, $m_2 = N/4$, $N/2$ popular object (popularity 4, $N/2$ non-popular objects (popularity 1).

For which system size N can mean-field be used?

Recall: $X_i^N(t)$ = fraction of objects in state i.

Theorem (Kurtz (1970s), Ying (2016)):

If the drift f is Lipschitz-continuous:	If in addition the ODE has a unique attractor π:
$$X^N(t) \approx x(t) + \frac{1}{\sqrt{N}} G_t$$	$$\mathbb{E}\left[X^N(\infty) - \pi\right] = O(1/\sqrt{N})$$



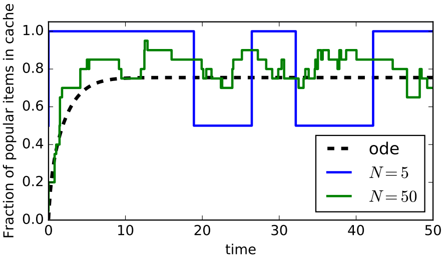
Example. Fraction of popular items in cache. $m_1 = N/4$, $m_2 = N/4$, $N/2$ popular object (popularity 4, $N/2$ non-popular objects (popularity 1).

For which system size N can mean-field be used?

Recall: $X_i^N(t) =$ fraction of objects in state i.

Theorem (Kurtz (1970s), Ying (2016)):

If the drift f is Lipschitz-continuous:	If in addition the ODE has a unique attractor π:
$$X^N(t) \approx x(t) + \frac{1}{\sqrt{N}} G_t$$	$$\mathbb{E}\left[X^N(\infty) - \pi\right] = O(1/\sqrt{N})$$



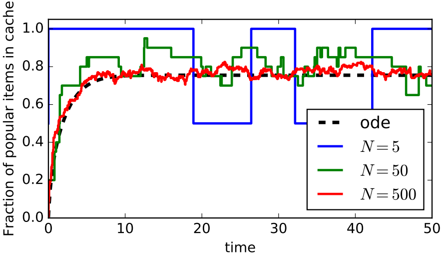
Example. Fraction of popular items in cache. $m_1 = N/4$, $m_2 = N/4$, $N/2$ popular object (popularity 4, $N/2$ non-popular objects (popularity 1).

For which system size N can mean-field be used?

Recall: $X_i^N(t) =$ fraction of objects in state i.

Theorem (Kurtz (1970s), Ying (2016)):

If the drift f is Lipschitz-continuous:	If in addition the ODE has a unique attractor π:
$$X^N(t) \approx x(t) + \frac{1}{\sqrt{N}} G_t$$	$$\mathbb{E}\left[X^N(\infty) - \pi\right] = O(1/\sqrt{N})$$



Example. Fraction of popular items in cache. $m_1 = N/4$, $m_2 = N/4$, $N/2$ popular object (popularity 4, $N/2$ non-popular objects (popularity 1).

In practice, mean field works well even for “small” systems

How small?

Cache size ($N/2$)	10	100	1000	$+\infty$
Proba popular$\in$cache	0.7604	0.7553	0.7548	0.7547
Error	0.0060	0.0006	$<$0.0001	–

Also works for load balancing policies (two-choice allocation):

N-servers	10	100	1000	$+\infty$
Average queue length (m^N)	3.81	3.39	3.36	3.35
Error ($m^N - m^\infty$)	0.45	0.039	0.004	0

In practice, mean field works well even for “small” systems

How small?

Cache size ($N/2$)	10	100	1000	$+\infty$
Proba popular$\in$cache	0.7604	0.7553	0.7548	0.7547
Error	0.0060	0.0006	$<$0.0001	–

Also works for load balancing policies (two-choice allocation):

N-servers	10	100	1000	$+\infty$
Average queue length (m^N)	3.81	3.39	3.36	3.35
Error ($m^N - m^\infty$)	0.45	0.039	0.004	0

X_i = fraction of objects in state i $\xleftrightarrow{O(1/\sqrt{N})}$ x_i (mean-field approx)

In practice, mean field works well even for "small" systems

How small?

Cache size ($N/2$)	10	100	1000	$+\infty$
Proba popular$\in$cache	0.7604	0.7553	0.7548	0.7547
Error	0.0060	0.0006	$<$0.0001	–

Also works for load balancing policies (two-choice allocation):

N-servers	10	100	1000	$+\infty$
Average queue length (m^N)	3.81	3.39	3.36	3.35
Error ($m^N - m^\infty$)	0.45	0.039	0.004	0

$O(1/\sqrt{N})$

X_i = fraction of objects in state i

x_i = Proba(one object is in i) = $\mathbb{E}[X_i]$

x_i (mean-field approx)

$O(1/\sqrt{N})$ (CLT)

?

Expected values estimated by mean field are 1/N-accurate

Theorem (Kolokoltsov 2012, G. 2017, G. and Van Houdt 2018). If the drift f is C^2 and has a unique exponentially stable attractor, then for any $t \in [0, \infty) \cup \{\infty\}$,

- there exists a (deterministic) vector $V(t)$ such that:

$$\mathbb{E}\left[X^N(t)\right] = x(t) + \frac{V(t)}{N} + O(1/N^2)$$

- $V(t)$ can be easily computed numerically

The refined mean field approximation...

... is defined as the classic mean field plus the $1/N$ correction term:

$$\mathbb{E}\left[X^N\right] = \underbrace{x(t) + \frac{V(t)}{N}}_{\text{refined mean field}} + O(1/N^2),$$

where $V(t)$ is computed analytically.

The refined mean field approximation...

... is defined as the classic mean field plus the $1/N$ correction term:

$$\mathbb{E}\left[X^N\right] = \underbrace{x(t) + \frac{V(t)}{N}}_{\text{refined mean field}} + O(1/N^2),$$

where $V(t)$ is computed analytically.

To compute $V(t)$, we need:

- Derivative of the drifts:

$$F^i_j(t) = \frac{\partial f_i}{\partial x_j}(x(t)) \text{ and } F^i_{jk}(t) = \frac{\partial^2 f_i}{\partial x_j \partial x_k}(x(t))$$

- A variance term:

$$Q(t) = \sum_{\ell} \ell \otimes \ell \beta_{\ell}(X(t))$$

Computational methods

Theorem (G, Van Houdt 2018, G. 2018) Given a density dependent process with twice-differentiable drift. Let $h : E \to \mathbb{R}$ be a twice-differentiable function, then for $t > 0$:

$$\mathbb{E}\left[X^N(t)\right] = x(t) + \frac{1}{N}V(t) + \frac{1}{N^2}A(t) + O(\frac{1}{N^3}),$$

where

$$\frac{d}{dt}V^i = \sum_j F^i_j V^j + \sum_{jk} F^i_{j,k} W^{j,k}$$

$$\frac{d}{dt}W^{j,k} = Q^{jk} + \sum_m F^j_m W^{m,k} + \sum_m W^{j,m} F^k_m$$

Theorem (G, Van Houdt 2018) The previous theorem also holds for the stationary regime ($t = +\infty$) if the ODE has a unique exponentially stable attractor.

Numerical example: caching and RANDOM(m) policy

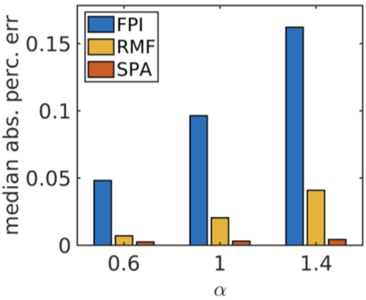
Cache size ($N/2$)	10	20	30	50	100	$+\infty$
Simulation	0.7604	0.7575	0.7566	0.7559	0.7553	–
Refined mf	0.7607	0.7576	0.7567	0.7558	0.7553	0.7547

Average fraction of most popular items in cache.

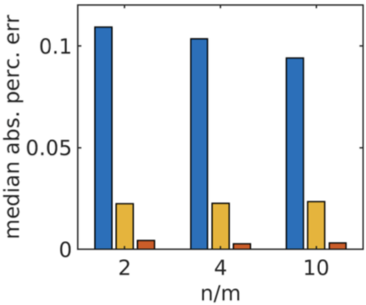
Numerical example: caching and RANDOM(m) policy

Cache size ($N/2$)	10	20	30	50	100	$+\infty$
Simulation	0.7604	0.7575	0.7566	0.7559	0.7553	–
Refined mf	0.7607	0.7576	0.7567	0.7558	0.7553	0.7547

Average fraction of most popular items in cache.

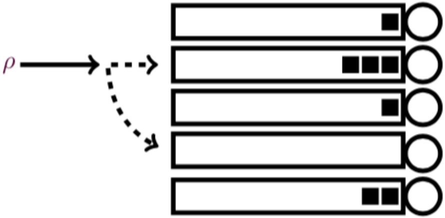


(a) $n = 10$



(b) $n = 10$

The SQ(2)



Arrival at each server ρ.

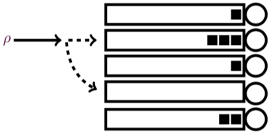
- Sample another queue.
- Allocate to the shortest of the two.

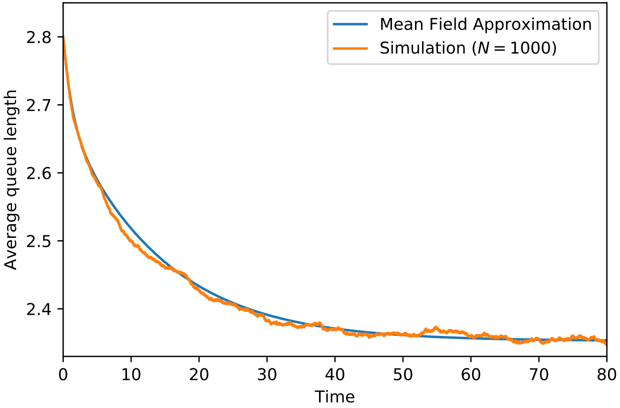
Service rate$=1$.

	$N = 10$	$N = 20$	$N = 50$	$N = 100$
Mean Field	2.3527	2.3527	2.3527	2.3527
$1/N$-expansion	2.7513	2.5520	2.4324	2.3925
$1/N^2$-expansion	2.8045	2.5653	2.4345	2.3930
Simulation	2.8003	2.5662	2.4350	2.3931

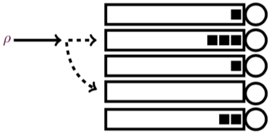
$SQ(2)$: **Steady-state average queue length** ($\rho = 0.9$).

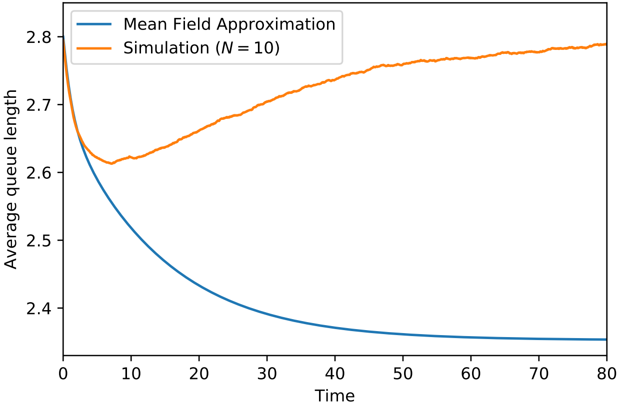
How does the expected queue length evolve with time?



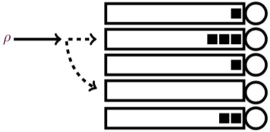


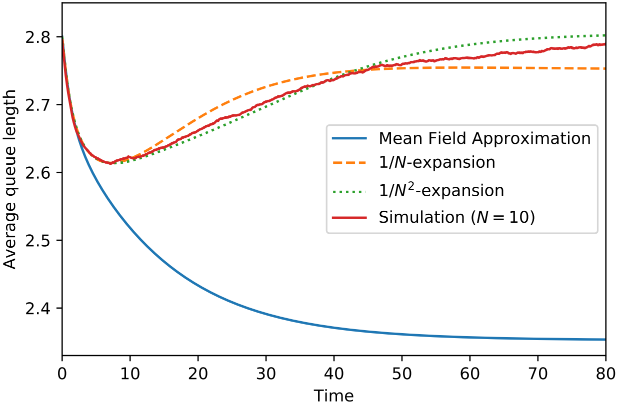
How does the expected queue length evolve with time?





How does the expected queue length evolve with time?



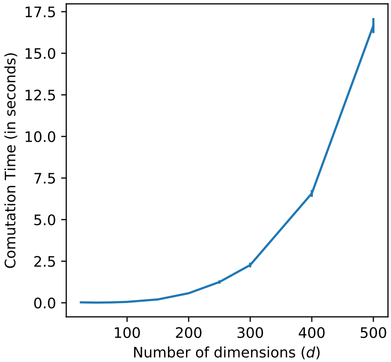


Remark about computation time :

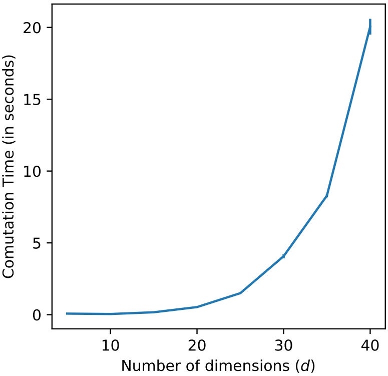
- 10min/1h (simulation $N = 1000/N = 10$), C++ code. Requires many simulations, confidence intervals,...
- 80ms (mean field), 700ms ($1/N$-expansion), 9s ($1/N^2$-expansion), Python `numpy`

Analysis of the computation time

For the numerical examples of $SQ(2)$, I used a bounded queue size d.



Time to compute the $1/N$-expansion



Time to compute the $1/N^2$-expansion

Analysis of the computation time (Python numpy implementation)

Does it always work?

Can I always exchange the limits $N \to \infty$, $k \to \infty$, $t \to \infty$?

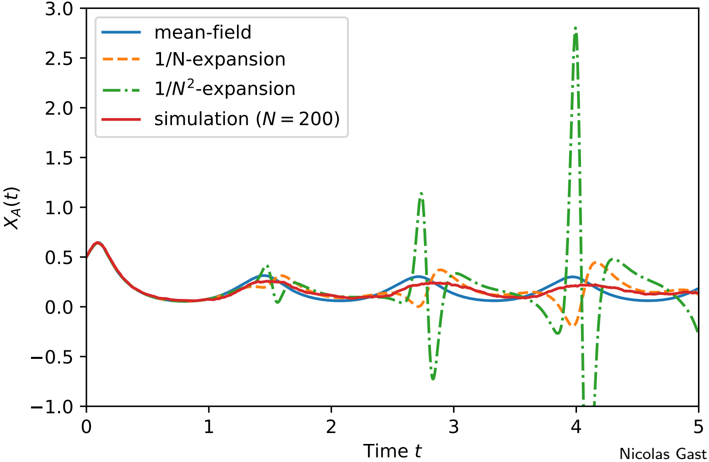
$$\mathbb{E}\left[X(t)\right] = x(t) + \frac{1}{N}V(t) + \frac{1}{N^2}A(t) + \dots + O(\frac{1}{N^{k+1}})$$

Does it always work?

Can I always exchange the limits $N \to \infty$, $k \to \infty$, $t \to \infty$?

$$\mathbb{E}\left[X(t)\right] = x(t) + \frac{1}{N}V(t) + \frac{1}{N^2}A(t) + \cdots + O(\frac{1}{N^{k+1}})$$

NO:



Outline

Where does V/N come from? ($SQ(2)$ example)

Transitions on X_i: $+\frac{1}{N}$ at rate $N(x_{i-1}^2 - x_i^2)$ and $-\frac{1}{N}$ at rate $N(x_i - x_{i+1})$.

Hence:

$$\begin{aligned}
\frac{d}{dt}\mathbb{E}\left[X_i(t)\right] &= \mathbb{E}\left[X_{i-1}^2(t) - X_i^2(t) - (X_i(t) - X_{i+1}(t))\right] \quad \text{(exact)}\\
&= \mathbb{E}\left[X_{i-1}^2(t)\right] - \mathbb{E}\left[X_i^2(t)\right] - \mathbb{E}\left[X_i(t)\right] + \mathbb{E}\left[X_{i+1}(t)\right]\\
&\approx \mathbb{E}\left[X_{i-1}(t)\right]^2 - \mathbb{E}\left[X_i(t)\right]^2 - \mathbb{E}\left[X_i(t)\right] + \mathbb{E}\left[X_{i+1}(t)\right] \quad \text{(mean field approx.}
\end{aligned}$$

Where does V/N come from? ($SQ(2)$ example)

Transitions on X_i: $+\frac{1}{N}$ at rate $N(x_{i-1}^2 - x_i^2)$ and $-\frac{1}{N}$ at rate $N(x_i - x_{i+1})$.

Hence:

$$\frac{d}{dt}\mathbb{E}\left[X_i(t)\right] = \mathbb{E}\left[X_{i-1}^2(t) - X_i^2(t) - (X_i(t) - X_{i+1}(t))\right] \quad \text{(exact)}$$

$$= \mathbb{E}\left[X_{i-1}^2(t)\right] - \mathbb{E}\left[X_i^2(t)\right] - \mathbb{E}\left[X_i(t)\right] + \mathbb{E}\left[X_{i+1}(t)\right]$$

$$\approx \cancel{\mathbb{E}\left[X_{i-1}(t)\right]^2 - \mathbb{E}\left[X_i(t)\right]^2 - \mathbb{E}\left[X_i(t)\right] + \mathbb{E}\left[X_{i+1}(t)\right]} \quad \text{(mean field approx.}$$

If we now consider how $\mathbb{E}\left[X_i^2\right]$ evolves, we have:

$$\frac{d}{dt}\mathbb{E}\left[X_i^2\right] = \mathbb{E}\left[(2X_i + \frac{1}{N})(X_{i-1}^2 - X_i^2) + (-2X_i + \frac{1}{N})(X_i - X_{i+1})\right]$$

$$= \mathbb{E}\left[\underbrace{2X_iX_{i-1}^2}_{\mathbb{E}\left[X_iX_{i-1}^2 \approx ?\right]} + \ldots\ldots\ldots\ldots\right]$$

where we denote X instead of $X(t)$ for simplicity.

To close the equation: we assume that $\mathbb{E}\left[(X_i - x_i)^2(X_{i-1} - x_{i-1})\right] \approx 0$.

System Size Expansion Approach

Recall that the transitions are $X \mapsto X + \frac{\ell}{N}$ at rate $N\beta_\ell(x)$.

$$\frac{d}{dt}\mathbb{E}\left[X\right] = \mathbb{E}\left[\sum_{\ell}\beta_\ell(X)\ell\right] = \mathbb{E}\left[f(X)\right] \qquad \text{(Exact)}$$

$$\frac{d}{dt}x = f(x) \qquad \text{(Mean Field Approx.)}$$

System Size Expansion Approach

Recall that the transitions are $X \mapsto X + \frac{\ell}{N}$ at rate $N\beta_\ell(x)$.

$$\frac{d}{dt}\mathbb{E}[X] = \mathbb{E}\left[\sum_\ell \beta_\ell(X)\ell\right] = \mathbb{E}[f(X)] \qquad \text{(Exact)}$$

$$\frac{d}{dt}x = f(x) \qquad \text{(Mean Field Approx.)}$$

We can now look at the second moment:

$$\begin{aligned}\mathbb{E}[(X-x)\otimes(X-x)] &= \mathbb{E}[(f(X)-f(x))\otimes(X-x)] \qquad \text{(Exact)}\\ &+ \mathbb{E}[(X-x)\otimes(f(X)-f(x))]\\ &+ \frac{1}{N}\mathbb{E}\left[\sum_{\ell\in\mathcal{L}}\beta_\ell(X)\ell\otimes\ell\right]\end{aligned}$$

System Size Expansion Approach

Recall that the transitions are $X \mapsto X + \frac{\ell}{N}$ at rate $N\beta_\ell(x)$.

$$\frac{d}{dt}\mathbb{E}[X] = \mathbb{E}\left[\sum_\ell \beta_\ell(X)\ell\right] = \mathbb{E}[f(X)] \qquad \text{(Exact)}$$

$$\frac{d}{dt}x = f(x) \qquad \text{(Mean Field Approx.)}$$

We can now look at the second moment:

$$\begin{aligned}\mathbb{E}[(X-x)\otimes(X-x)] &= \mathbb{E}[(f(X)-f(x))\otimes(X-x)] \qquad \text{(Exact)}\\ &+ \mathbb{E}[(X-x)\otimes(f(X)-f(x))]\\ &+ \frac{1}{N}\mathbb{E}\left[\sum_{\ell\in\mathcal{L}}\beta_\ell(X)\ell\otimes\ell\right]\end{aligned}$$

... We can also look at higher order moments

$$\begin{aligned}\mathbb{E}\left[(X-x)^{\otimes 3}\right] &= 3\mathrm{Sym}\mathbb{E}[(f(X)-f(x))\otimes(X-x)\otimes(X-x)]\\ &+ \frac{3}{N}\mathrm{Sym}\mathbb{E}\left[\sum_{\ell\in\mathcal{L}}\beta_\ell(X)\ell\otimes\ell\otimes(X-x)\right] + \frac{1}{N}\mathbb{E}\left[\sum_{\ell\in\mathcal{L}}\beta_\ell(X)\ell\otimes\ell\otimes\ell\right]\end{aligned}$$

System Size Expansion and Moment Closure

Let $x(t)$ be the mean field approximation and $Y(t) = X(t) - x(t)$, and $Y(t)^{(k)} = \underbrace{Y(t) \otimes \cdots \otimes Y(t)}_{k \text{ times}}$

$\frac{d}{dt}\mathbb{E}\left[Y(t)^{(k)}\right]$ can be expressed as **an exact** function of $Y(t)^{(j)}$ for $j \in \{0 \ldots, k+1\}$.

System Size Expansion and Moment Closure

Let $x(t)$ be the mean field approximation and $Y(t) = X(t) - x(t)$, and $Y(t)^{(k)} = \underbrace{Y(t) \otimes \cdots \otimes Y(t)}_{k \text{ times}}$

$\frac{d}{dt}\mathbb{E}\left[Y(t)^{(k)}\right]$ can be expressed as **an exact** function of $Y(t)^{(j)}$ for $j \in \{0\ldots, k+1\}$.

You can close the equations by assuming that $Y^{(k)} \approx 0$ for $k \geq K$.

- For $K = 1$, this gives the mean field approximation ($1/N$-accurate)
- For $K = 3$, this gives the refined mean field ($1/N^2$-accurate).
- For $K = 5$, this gives a second order expansion ($1/N^3$-accurate).

Limit: For a system of dimension d, $Y(t)^{(k)}$ has d^k equations.

Outline

Recap and extensions

For a mean field model with twice differentiable drift, then :

1. The accuracy of the classical mean field approximation is $O(1/N)$.
2. We can use this to define a refined approximation.
3. The refined approximation is often accurate for $N = 10$.

Extensions:

- Transient regime
- Discrete-time (Synchronous)
- Next expansion term in $1/N^2$.

In many cases, the refined approximation is very accurate

"Truth"	Refined mean field approximation	Mean field approximation
$\mathbb{E}\left[X^N\right]$	$\pi + \dfrac{V}{N}$	π (=fixed point)

	Coupon	Supermarket	Pull/push
Simulation ($N = 10$)	1.530	2.804	2.304
Refined mean field ($N = 10$)	1.517	2.751	2.295
Mean field ($N = \infty$)	1.250	2.353	1.636

[6]

[6] Ref : G., Van Houdt, 2018

Some open questions

- Optimization (mean field games, reinforcement learning)
- Non-exponential models (ex: general service time, TTL&LRU)
- Heterogeneous models
- Multi-timescale models

Some References

http://mescal.imag.fr/membres/nicolas.gast

nicolas.gast@inria.fr

https://github.com/ngast/rmf_tool/

- A Refined Mean Field Approximation by Gast and Van Houdt. SIGMETRICS 2018 (best paper award)
- Size Expansions of Mean Field Approximation: Transient and Steady-State Analysis Gast, Bortolussi, Tribastone
- Expected Values Estimated via Mean Field Approximation are O(1/N)-accurate by Gast. SIGMETRICS 2017.