Learning traffic correlations in multi-class queueing systems by sampling workloads

Martin Zubeldia
(Georgia Tech)

Joint work with *Michel Mandjes* (U. of Amsterdam)

September 21-22, 2021
Data-Driven Queueing Challenges workshop

Motivation: Backbone of the Internet

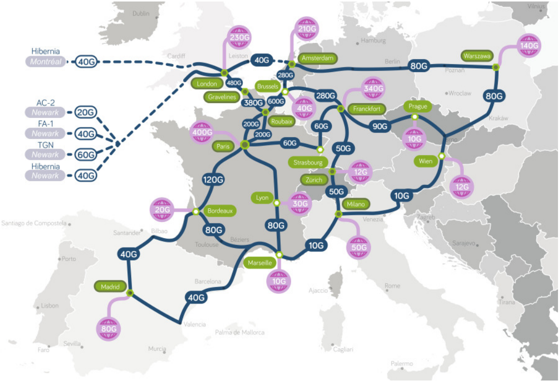


Figure: OVH Europe network

Motivation: Backbone of the Internet

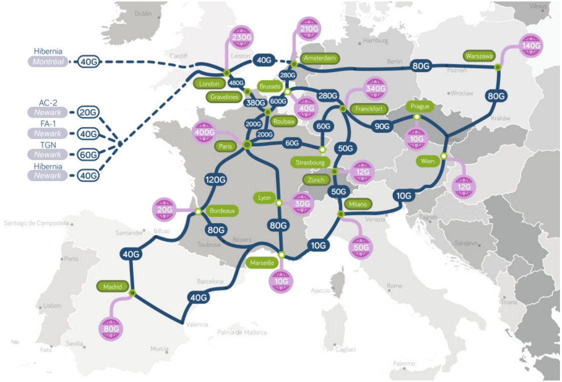


Figure: OVH Europe network

Features:

- **Traffic:** Highly aggregated

Motivation: Backbone of the Internet

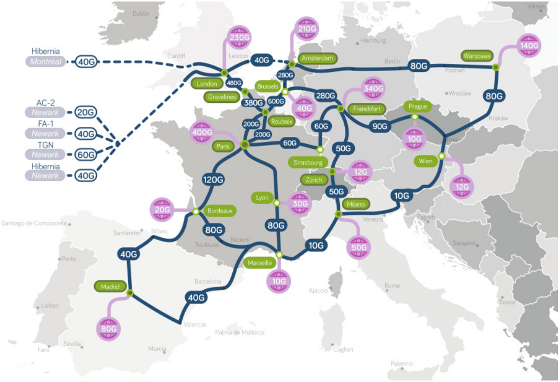
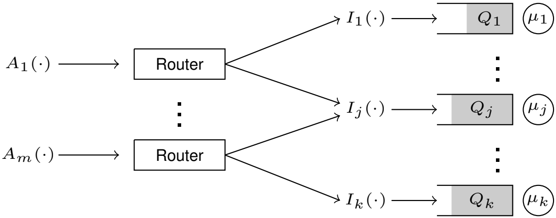


Figure: OVH Europe network

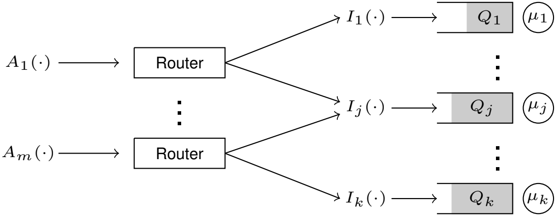
Features:

- **Traffic:** Highly aggregated
- **Routing:** Mostly static

Stylized model: Static load balancing



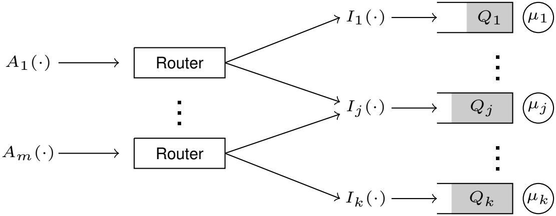
Stylized model: Static load balancing



Assumptions:

- **Arrivals:** $A(\cdot)$ is Gaussian with known rate $\lambda \in \mathbb{R}_+^m$ and unknown covariance matrix $\Sigma : \mathbb{R}^2 \to \mathbb{R}^{m \times m}$

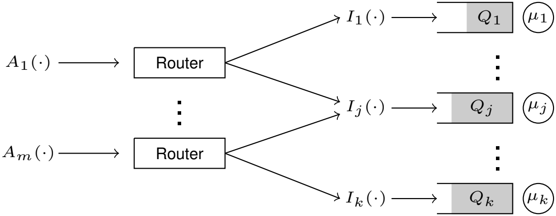
Stylized model: Static load balancing



Assumptions:

- **Arrivals:** $A(\cdot)$ is Gaussian with known rate $\lambda \in \mathbb{R}_+^m$ and unknown covariance matrix $\Sigma : \mathbb{R}^2 \to \mathbb{R}^{m \times m}$
- **Routing:** Static deterministic split with routing matrix R

Stylized model: Static load balancing



Assumptions:

- **Arrivals:** $A(\cdot)$ is Gaussian with known rate $\lambda \in \mathbb{R}_+^m$ and unknown covariance matrix $\Sigma : \mathbb{R}^2 \to \mathbb{R}^{m \times m}$
- **Routing:** Static deterministic split with routing matrix R

Objective: Learn

$$R^* \in \arg\min \left\{ \max_{i \in \{1,\ldots,k\}} \left\{ \mathbb{P}\big(Q_i(R) > b_i\big) \right\} \right\}$$

Additional assumptions on $A(\cdot)$

Assumption: Many-sources regime

$$A(\cdot) = A^{(n)} = \frac{1}{n} \sum_{i=1}^{n} X^{(i)}(\cdot),$$

where $\left\{ X^{(i)}(\cdot) \right\}_{i \geq 1}$ are i.i.d.

Additional assumptions on $A(\cdot)$

Assumption: Many-sources regime

$$A(\cdot) = A^{(n)} = \frac{1}{n}\sum_{i=1}^{n} X^{(i)}(\cdot),$$

where $\left\{X^{(i)}(\cdot)\right\}_{i\geq 1}$ are i.i.d.

For simplicity: Consider multivariate fractional Brownian motions (mfBm)

$$Cov\left(A_i^{(n)}(t),\ A_j^{(n)}(s)\right) = \frac{\sigma_i\sigma_j\rho_{i,j}}{2n}\left(|t|^{2H} + |s|^{2H} - |s-t|^{2H}\right)$$

- Hurst parameter: $H \in (0,1)$
- Variance: $\sigma_i^2 > 0$
- Correlation: $\rho_{i,j} \in [-1,\ 1]$

How do we find the optimal routing matrix?

Approach 1: directly learn model parameters

Algorithm:

- Observe $A^{(n)}(\cdot)$ and estimate covariance matrix

Approach 1: directly learn model parameters

Algorithm:

- Observe $A^{(n)}(\cdot)$ and estimate covariance matrix
- Use large-deviations approximation to get $\mathbb{P}\left(Q_i^{(n)}(R) > b_i\right)$ for any R

Approach 1: directly learn model parameters

Algorithm:

- Observe $A^{(n)}(\cdot)$ and estimate covariance matrix
- Use large-deviations approximation to get $\mathbb{P}\left(Q_i^{(n)}(R) > b_i\right)$ for any R
- Solve optimization problem for R^*

Approach 1: directly learn model parameters

Algorithm:

- Observe $A^{(n)}(\cdot)$ and estimate covariance matrix
- Use large-deviations approximation to get $\mathbb{P}\left(Q_i^{(n)}(R) > b_i\right)$ for any R
- Solve optimization problem for R^*

Advantage:

- Only one estimation instance

Approach 1: directly learn model parameters

Algorithm:

- Observe $A^{(n)}(\cdot)$ and estimate covariance matrix
- Use large-deviations approximation to get $\mathbb{P}\left(Q_i^{(n)}(R) > b_i\right)$ for any R
- Solve optimization problem for R^*

Advantage:

- Only one estimation instance

Disadvantage:

- It is hard to estimate covariance matrix

Approach 2: directly learn optimal routing

Algorithm:

- Start with initial routing matrix R_0

Approach 2: directly learn optimal routing

Algorithm:

- Start with initial routing matrix R_0
- Observe queues and estimate $\mathbb{P}\left(Q_i^{(n)}(R_0) > b_i\right)$

Approach 2: directly learn optimal routing

Algorithm:

- Start with initial routing matrix R_0
- Observe queues and estimate $\mathbb{P}\left(Q_i^{(n)}(R_0) > b_i\right)$
- Iterate forming a sequence that converges to R^*

Approach 2: directly learn optimal routing

Algorithm:

- Start with initial routing matrix R_0
- Observe queues and estimate $\mathbb{P}\left(Q_i^{(n)}(R_0) > b_i\right)$
- Iterate forming a sequence that converges to R^*

Advantage:

- Queue lengths are easier to estimate

Approach 2: directly learn optimal routing

Algorithm:

- Start with initial routing matrix R_0
- Observe queues and estimate $\mathbb{P}\left(Q_i^{(n)}(R_0) > b_i\right)$
- Iterate forming a sequence that converges to R^*

Advantage:

- Queue lengths are easier to estimate

Disadvantage:

- Possibly slow to converge

Optimization with indirect learning

Algorithm:

- Start with initial routing matrix R_0

Optimization with indirect learning

Algorithm:

- Start with initial routing matrix R_0
- Observe queues and estimate steady-state $Q^{(n)}(R_0)$

Optimization with indirect learning

Algorithm:

- Start with initial routing matrix R_0
- Observe queues and estimate steady-state $Q^{(n)}(R_0)$
- **Key:** Use an inversion procedure to obtain covariance matrix

Optimization with indirect learning

Algorithm:

- Start with initial routing matrix R_0
- Observe queues and estimate steady-state $Q^{(n)}(R_0)$
- **Key:** Use an inversion procedure to obtain covariance matrix
- Use large-deviations approximation to get $\mathbb{P}\left(Q_i^{(n)}(R) > b_i\right)$ for any R

Optimization with indirect learning

Algorithm:

- Start with initial routing matrix R_0
- Observe queues and estimate steady-state $Q^{(n)}(R_0)$
- **Key:** Use an inversion procedure to obtain covariance matrix
- Use large-deviations approximation to get $\mathbb{P}\left(Q_i^{(n)}(R) > b_i\right)$ for any R
- Solve optimization problem for R^*

Optimization with indirect learning

Algorithm:

- Start with initial routing matrix R_0
- Observe queues and estimate steady-state $Q^{(n)}(R_0)$
- **Key:** Use an inversion procedure to obtain covariance matrix
- Use large-deviations approximation to get $\mathbb{P}\left(Q_i^{(n)}(R) > b_i\right)$ for any R
- Solve optimization problem for R^*

Advantages:

- Queue lengths are easier to estimate than covariances
- Fast convergence

First inversion procedure

From marginal queue lengths to variances

[Mandjes & van de Meent (2009) Resource dimensioning through buffer sampling]

From marginal queue lengths to variances

[Mandjes & van de Meent (2009) Resource dimensioning through buffer sampling]

Work flow:

$$A^{(n)}(\cdot) \xrightarrow{Routing} I_i^{(n)}(\cdot) \xrightarrow{Queueing} Q_i^{(n)}$$

From marginal queue lengths to variances

[Mandjes & van de Meent (2009) Resource dimensioning through buffer sampling]

Work flow:

$$A^{(n)}(\cdot) \xrightarrow{Routing} I_i^{(n)}(\cdot) \xrightarrow{Queueing} Q_i^{(n)}$$

Inversion:

$$Q_i^{(n)} \xrightarrow{\text{``}Inversion\text{''}} Var\left(I_i^{(n)}(\cdot)\right)$$

From marginal queue lengths to variances

[Mandjes & van de Meent (2009) Resource dimensioning through buffer sampling]

Work flow:

$$A^{(n)}(\cdot) \xrightarrow{Routing} I_i^{(n)}(\cdot) \xrightarrow{Queueing} Q_i^{(n)}$$

Inversion:

$$Q_i^{(n)} \xrightarrow{\text{“}Inversion\text{”}} Var\left(I_i^{(n)}(\cdot)\right)$$

based on the large-deviations principle

$$\lim_{n\to\infty} -\frac{1}{n}\log\left(\mathbb{P}\left(Q_i^{(n)} > b\right)\right) = \inf_{t<0}\left\{\frac{\left[b-(\mu_i-\overline{\lambda}_i)t\right]^2}{2nVar\left(I_i^{(n)}(t)\right)}\right\}$$

From marginal queue lengths to variances

$$\lim_{n\to\infty} -\frac{1}{n} \log\left(\mathbb{P}\left(Q_i^{(n)} > b\right)\right) \leq \frac{\left[b - (\mu_i - \overline{\lambda}_i)t\right]^2}{2nVar\left(I_i^{(n)}(t)\right)}$$

From marginal queue lengths to variances

$$\lim_{n\to\infty} -\frac{1}{n}\log\left(\mathbb{P}\left(Q_i^{(n)} > b\right)\right) \le \frac{\left[b-(\mu_i-\overline{\lambda}_i)t\right]^2}{2nVar\left(I_i^{(n)}(t)\right)}$$

$$nVar\left(I_i^{(n)}(t)\right) \le \frac{\left[b-(\mu_i-\overline{\lambda}_i)t\right]^2}{-2\lim_{n\to\infty}\frac{1}{n}\log\left(\mathbb{P}\left(Q_i^{(n)} > b\right)\right)}$$

From marginal queue lengths to variances

$$\lim_{n\to\infty} -\frac{1}{n} \log\left(\mathbb{P}\left(Q_i^{(n)} > b\right)\right) \leq \frac{\left[b - (\mu_i - \overline{\lambda}_i)t\right]^2}{2nVar\left(I_i^{(n)}(t)\right)}$$

$$nVar\left(I_i^{(n)}(t)\right) \leq \frac{\left[b - (\mu_i - \overline{\lambda}_i)t\right]^2}{-2\lim_{n\to\infty} \frac{1}{n} \log\left(\mathbb{P}\left(Q_i^{(n)} > b\right)\right)}$$

$$nVar\left(I_i^{(n)}(t)\right) \leq \inf_{b>0} \left\{ \frac{\left[b - (\mu_i - \overline{\lambda}_i)t\right]^2}{-2\lim_{n\to\infty} \frac{1}{n} \log\left(\mathbb{P}\left(Q_i^{(n)} > b\right)\right)} \right\}$$

From marginal queue lengths to variances

$$\lim_{n\to\infty} -\frac{1}{n}\log\left(\mathbb{P}\left(Q_i^{(n)} > b\right)\right) \leq \frac{\left[b-(\mu_i-\overline{\lambda}_i)t\right]^2}{2nVar\left(I_i^{(n)}(t)\right)}$$

$$nVar\left(I_i^{(n)}(t)\right) \leq \frac{\left[b-(\mu_i-\overline{\lambda}_i)t\right]^2}{-2\lim_{n\to\infty}\frac{1}{n}\log\left(\mathbb{P}\left(Q_i^{(n)} > b\right)\right)}$$

$$nVar\left(I_i^{(n)}(t)\right) \leq \inf_{b>0}\left\{\frac{\left[b-(\mu_i-\overline{\lambda}_i)t\right]^2}{-2\lim_{n\to\infty}\frac{1}{n}\log\left(\mathbb{P}\left(Q_i^{(n)} > b\right)\right)}\right\}$$

$$Var\left(I_i^{(n)}(t)\right) \lessapprox \inf_{b\in[\epsilon,1/\epsilon]}\left\{\frac{\left[b-(\mu_i-\overline{\lambda}_i)t\right]^2}{-2\log\left(\hat{\mathbb{P}}_N\left(Q_i^{(n)} > b\right)\right)}\right\}$$

From marginal queue lengths to variances

Variance estimator:

$$\hat{V}_{i,\epsilon}^{(n,N)}(t) \triangleq \inf_{b\in[\epsilon,1/\epsilon]} \left\{ \frac{\left[b-(\mu_i-\overline{\lambda}_i)t\right]^2}{-2\log\left(\hat{\mathbb{P}}_N\left(Q_i^{(n)}>b\right)\right)} \right\} \overset{?}{\approx} Var\left(I_i^{(n)}(t)\right)$$

From marginal queue lengths to variances

Variance estimator:

$$\hat{V}_{i,\epsilon}^{(n,N)}(t) \triangleq \inf_{b \in [\epsilon, 1/\epsilon]} \left\{ \frac{\left[b - (\mu_i - \overline{\lambda}_i)t\right]^2}{-2 \log\left(\hat{\mathbb{P}}_N\left(Q_i^{(n)} > b\right)\right)} \right\} \overset{?}{\approx} Var\left(I_i^{(n)}(t)\right)$$

Theorem

Fix $t < 0$. We have

$$\lim_{n \to \infty} \lim_{N \to \infty} n \left| \hat{V}_{i,\epsilon}^{(n,N)}(t) - Var\left(I_i^{(n)}(t)\right) \right| = 0, \quad a.s.,$$

for all ϵ small enough.

From marginal queue lengths to variances

- Need to find the $m(m+1)/2$ distinct $Cov\left(A_i^{(n)}(\cdot),\, A_j^{(n)}(\cdot)\right)$

From marginal queue lengths to variances

- Need to find the $m(m+1)/2$ distinct $Cov\left(A_i^{(n)}(\cdot),\ A_j^{(n)}(\cdot)\right)$

- For each of the k queues we obtain a linear equation

$$\hat{V}_{i,\epsilon}^{(n,N)}(\cdot) \approx Var\left(I_j^{(n)}(\cdot)\right) = \sum_{j=1}^{m}\sum_{q=1}^{m} R_{j,i}R_{q,i}Cov\left(A_j^{(n)}(\cdot),\ A_q^{(n)}(\cdot)\right)$$

From marginal queue lengths to variances

- Need to find the $m(m+1)/2$ distinct $Cov\left(A_i^{(n)}(\cdot),\, A_j^{(n)}(\cdot)\right)$

- For each of the k queues we obtain a linear equation

$$\hat{V}_{i,\epsilon}^{(n,N)}(\cdot) \approx Var\left(I_j^{(n)}(\cdot)\right) = \sum_{j=1}^{m}\sum_{q=1}^{m} R_{j,i}R_{q,i}Cov\left(A_j^{(n)}(\cdot),\; A_q^{(n)}(\cdot)\right)$$

- Repeating this for different R yields enough l.i. equations

From marginal queue lengths to variances

- Need to find the $m(m+1)/2$ distinct $Cov\left(A_i^{(n)}(\cdot),\, A_j^{(n)}(\cdot)\right)$

- For each of the k queues we obtain a linear equation

$$\hat{V}_{i,\epsilon}^{(n,N)}(\cdot) \approx Var\left(I_j^{(n)}(\cdot)\right) = \sum_{j=1}^{m}\sum_{q=1}^{m} R_{j,i}R_{q,i}Cov\left(A_j^{(n)}(\cdot),\; A_q^{(n)}(\cdot)\right)$$

- Repeating this for different R yields enough l.i. equations

Use joint queue lengths to get more equations directly?

Second inversion procedure

From pair-wise joint queue lengths to covariances

Work flow:

$$A^{(n)}(\cdot) \xrightarrow{Routing} \left(I_i^{(n)}(\cdot),\, I_j^{(n)}(\cdot)\right) \xrightarrow{Queueing} \left(Q_i^{(n)},\, Q_j^{(n)}\right)$$

From pair-wise joint queue lengths to covariances

Work flow:

$$A^{(n)}(\cdot) \xrightarrow{Routing} \left(I_i^{(n)}(\cdot),\, I_j^{(n)}(\cdot)\right) \xrightarrow{Queueing} \left(Q_i^{(n)},\, Q_j^{(n)}\right)$$

Inversion:

$$\left(Q_i^{(n)},\, Q_j^{(n)}\right) \xrightarrow{\text{``}Inversion\text{''}} Cov\left(I_i^{(n)}(\cdot),\, I_j^{(n)}(\cdot)\right)$$

From pair-wise joint queue lengths to covariances

Work flow:

$$A^{(n)}(\cdot) \xrightarrow{Routing} \left(I_i^{(n)}(\cdot),\, I_j^{(n)}(\cdot)\right) \xrightarrow{Queueing} \left(Q_i^{(n)},\, Q_j^{(n)}\right)$$

Inversion:

$$\left(Q_i^{(n)},\, Q_j^{(n)}\right) \xrightarrow{\text{``}Inversion\text{''}} Cov\left(I_i^{(n)}(\cdot),\, I_j^{(n)}(\cdot)\right)$$

based on the large-deviations principle

$$\lim_{n\to\infty} -\frac{1}{n}\log\left(\mathbb{P}\left(c_i Q_i^{(n)} + c_j Q_j^{(n)} > 1\right)\right) = \inf_{t,s<0}\left\{\frac{\left[1 - c_i\left(\mu_i - \overline{\lambda}_i\right)t - c_j\left(\mu_j - \overline{\lambda}_j\right)s\right]^2}{2Var\left(c_i I_i^{(n)}(t) + c_j I_j^{(n)}(s)\right)}\right\}$$

Feasibility of the inversion

Covariance estimator:

$$\hat{C}_{i,j,\epsilon}^{(n,N)}(t,s) \triangleq \inf_{c_i,c_j \in [\epsilon, 1/\epsilon]} \left\{ \cdots\cdots\cdots\cdots \right\} \overset{?}{\approx} Cov\left(I_i^{(n)}(t),\, I_j^{(n)}(s)\right)$$

Feasibility of the inversion

Covariance estimator:

$$\hat{C}_{i,j,\epsilon}^{(n,N)}(t,s) \triangleq \inf_{c_i,c_j \in [\epsilon, 1/\epsilon]} \left\{ \cdots\cdots\cdots\cdots \right\} \overset{?}{\approx} Cov\left(I_i^{(n)}(t),\, I_j^{(n)}(s) \right)$$

Theorem

If $I^{(n)}(\cdot)$ is short-range dependent ($H \leq 1/2$), and $I_i^{(n)}(\cdot)$ and $I_j^{(n)}(\cdot)$ are non-negatively correlated ($\rho_{i,j} \geq 0$), then

$$\lim_{n\to\infty} \lim_{N\to\infty} n \left| \hat{C}_{i,j,\epsilon}^{(n,N)}(t,t) - Cov\left(I_i^{(n)}(t),\, I_j^{(n)}(t) \right) \right| = 0, \quad a.s.,$$

for all ϵ small enough.

Feasibility of the inversion

- Need to find the $m(m+1)/2$ distinct $Cov\left(A_i^{(n)}(\cdot),\, A_j^{(n)}(\cdot)\right)$

Feasibility of the inversion

- Need to find the $m(m+1)/2$ distinct $Cov\left(A_i^{(n)}(\cdot),\ A_j^{(n)}(\cdot)\right)$
- For each of the k queues we obtain a linear equation

$$\hat{V}_{i,\epsilon}^{(n,N)}(\cdot) \approx \sum_{j=1}^{m}\sum_{q=1}^{m} R_{j,i}R_{q,i}Cov\left(A_j^{(n)}(\cdot),\ A_q^{(n)}(\cdot)\right)$$

Feasibility of the inversion

- Need to find the $m(m+1)/2$ distinct $Cov\left(A_i^{(n)}(\cdot),\, A_j^{(n)}(\cdot)\right)$
- For each of the k queues we obtain a linear equation

$$\hat{V}_{i,\epsilon}^{(n,N)}(\cdot) \approx \sum_{j=1}^{m}\sum_{q=1}^{m} R_{j,i}R_{q,i}Cov\left(A_j^{(n)}(\cdot),\; A_q^{(n)}(\cdot)\right)$$

- For each of the $k(k-1)/2$ pairs (i,j), with $i<j$, we get

$$\hat{C}_{i,j,\epsilon}^{(n,N)}(\cdot,\cdot) \approx \sum_{q=1}^{m}\sum_{l=1}^{m} R_{q,i}R_{l,j}Cov\left(A_j^{(n)}(\cdot),\; A_q^{(n)}(\cdot)\right)$$

Feasibility of the inversion

- Need to find the $m(m+1)/2$ distinct $Cov\left(A_i^{(n)}(\cdot),\ A_j^{(n)}(\cdot)\right)$
- For each of the k queues we obtain a linear equation

$$\hat{V}_{i,\epsilon}^{(n,N)}(\cdot) \approx \sum_{j=1}^{m}\sum_{q=1}^{m} R_{j,i}R_{q,i}Cov\left(A_j^{(n)}(\cdot),\ A_q^{(n)}(\cdot)\right)$$

- For each of the $k(k-1)/2$ pairs (i,j), with $i<j$, we get

$$\hat{C}_{i,j,\epsilon}^{(n,N)}(\cdot,\cdot) \approx \sum_{q=1}^{m}\sum_{l=1}^{m} R_{q,i}R_{l,j}Cov\left(A_j^{(n)}(\cdot),\ A_q^{(n)}(\cdot)\right)$$

- If $m \leq k$, these are enough for a single R

Fundamental limitation

Theorem

If $I^{(n)}(\cdot)$ is long-range dependent ($H > 1/2$), and $I_i^{(n)}(\cdot)$ and $I_j^{(n)}(\cdot)$ are negatively correlated ($\rho_{i,j} < 0$), then $Cov\left(I_i^{(n)}(t),\, I_j^{(n)}(t)\right)$ **cannot be recovered** from the large deviations behavior of $c_i Q_i^{(n)} + c_j Q_j^{(n)}$.

There is an inherent loss of information!

Why we have this limitation?

Why we have this limitation?

Three explanations:

- The analytical one
- Sample path large deviations interpretation
- Intuitive with queues as reflected processes

Analytical explanation

Recall:

$$-\log\left(\mathbb{P}\left(c_iQ_i^{(n)}+c_jQ_j^{(n)}>1\right)\right)$$

$$\approx \inf_{t,s<0}\left\{\frac{\left[1-c_i\left(\mu_i-\overline{\lambda}_i\right)t-c_j\left(\mu_j-\overline{\lambda}_j\right)s\right]^2}{2Var\left(c_iI_i^{(n)}(t)+c_jI_j^{(n)}(s)\right)}\right\}$$

Analytical explanation

Recall:

$$-\log\left(\mathbb{P}\left(c_iQ_i^{(n)}+c_jQ_j^{(n)}>1\right)\right) \approx \inf_{t,s<0}\left\{\frac{\left[1-c_i\left(\mu_i-\overline{\lambda}_i\right)t-c_j\left(\mu_j-\overline{\lambda}_j\right)s\right]^2}{2Var\left(c_iI_i^{(n)}(t)+c_jI_j^{(n)}(s)\right)}\right\}$$

We show: If (t^*, s^*) is a minimizer $\Rightarrow$ either $t^*=0$ or $s^*=0$, and

$$-\log\left(\mathbb{P}\left(c_iQ_i^{(n)}+c_jQ_j^{(n)}>1\right)\right) \approx \min\left\{\inf_{t<0}\left\{\frac{\left[1-c_i\left(\mu_i-\overline{\lambda}_i\right)t\right]^2}{2c_i^2Var\left(I_i^{(n)}(t)\right)}\right\},\inf_{s<0}\left\{\frac{\left[1-c_j\left(\mu_j-\overline{\lambda}_j\right)s\right]^2}{2c_j^2Var\left(I_j^{(n)}(s)\right)}\right\}\right\}$$

Sample path large deviations interpretation

Let: $q_i(\cdot)$ and $q_j(\cdot)$ be most likely paths for the event

$$\left\{ c_i Q_i^{(n)}(0) + c_j Q_j^{(n)}(0) = 1 \right\}$$

Sample path large deviations interpretation

Let: $q_i(\cdot)$ and $q_j(\cdot)$ be most likely paths for the event

$$\left\{ c_i Q_i^{(n)}(0) + c_j Q_j^{(n)}(0) = 1 \right\}$$

Note: The minimizers (t^*, s^*) are such that

$t^* = \sup\{t < 0 : q_i(t) = 0\}$ and $s^* = \sup\{s < 0 : q_j(s) = 0\}$

Sample path large deviations interpretation

Let: $q_i(\cdot)$ and $q_j(\cdot)$ be most likely paths for the event

$$\left\{c_i Q_i^{(n)}(0) + c_j Q_j^{(n)}(0) = 1\right\}$$

Note: The minimizers (t^*, s^*) are such that

$t^* = \sup\{t < 0 : q_i(t) = 0\}$ and $s^* = \sup\{s < 0 : q_j(s) = 0\}$

Here: Since either $t^* = 0$ or $s^* = 0$, most likely paths are in

$$\left\{c_i Q_i^{(n)}(0) = 1 \text{ and } c_j Q_j^{(n)}(0) = 0\right\} \cup \left\{c_i Q_i^{(n)}(0) = 0 \text{ and } c_j Q_j^{(n)}(0) = 1\right\}$$

Intuition with queues as reflected processes

Note: $Q_i^{(n)}(\cdot)$ and $Q_j^{(n)}(\cdot)$ are the reflection at 0 of

$$W_i^{(n)}(t) \triangleq I_i^{(n)}(t) - \mu_i t \qquad \text{and} \qquad W_j^{(n)}(t) \triangleq I_j^{(n)}(t) - \mu_j t$$

Intuition with queues as reflected processes

Note: $Q_i^{(n)}(\cdot)$ and $Q_j^{(n)}(\cdot)$ are the reflection at 0 of

$$W_i^{(n)}(t) \triangleq I_i^{(n)}(t) - \mu_i t \qquad \text{and} \qquad W_j^{(n)}(t) \triangleq I_j^{(n)}(t) - \mu_j t$$

In this case: Because of negative correlation:

- When $I_i^{(n)}(\cdot)$ grows faster, $I_j^{(n)}(\cdot)$ grows slower

Intuition with queues as reflected processes

Note: $Q_i^{(n)}(\cdot)$ and $Q_j^{(n)}(\cdot)$ are the reflection at 0 of

$$W_i^{(n)}(t) \triangleq I_i^{(n)}(t) - \mu_i t \qquad \text{and} \qquad W_j^{(n)}(t) \triangleq I_j^{(n)}(t) - \mu_j t$$

In this case: Because of negative correlation:

- When $I_i^{(n)}(\cdot)$ grows faster, $I_j^{(n)}(\cdot)$ grows slower
- $Q_i^{(n)}(\cdot)$ increases and $Q_j^{(n)}(\cdot)$ decreases (until it's reflected)

Intuition with queues as reflected processes

Note: $Q_i^{(n)}(\cdot)$ and $Q_j^{(n)}(\cdot)$ are the reflection at 0 of

$$W_i^{(n)}(t) \triangleq I_i^{(n)}(t) - \mu_i t \qquad \text{and} \qquad W_j^{(n)}(t) \triangleq I_j^{(n)}(t) - \mu_j t$$

In this case: Because of negative correlation:

- When $I_i^{(n)}(\cdot)$ grows faster, $I_j^{(n)}(\cdot)$ grows slower
- $Q_i^{(n)}(\cdot)$ increases and $Q_j^{(n)}(\cdot)$ decreases (until it's reflected)
- Magnitude of correlation is lost in the reflection at 0

Back to finding the optimal routing matrix

Estimating the overflow probabilities

- We have the estimates

$$\hat{\mathscr{C}}_{\epsilon,j,\ell}^{(n,N)}(t,t) \approx \mathbb{C}\mathrm{ov}\left(A_j^{(n)}(t),\, A_\ell^{(n)}(t)\right),$$

Estimating the overflow probabilities

- We have the estimates

$$\hat{\mathscr{C}}_{\epsilon,j,\ell}^{(n,N)}(t,t) \approx \mathbb{C}\text{ov}\left(A_j^{(n)}(t),\ A_\ell^{(n)}(t)\right),$$

- For any matrix R, we approximate $\mathbb{P}\left(Q_i^{(n)}(R) > b_i\right)$ as

$$\hat{P}_{\epsilon,\delta,i}^{(n,N)}(R) \triangleq \exp\left(-\inf_{t\in[-1/\delta,-\delta]}\left\{\frac{\left[b_i - \left(\mu_i - \sum_{j=1}^{m} R_{j,i}\lambda_j\right)t\right]^2}{\sum_{j=1}^{m}\sum_{\ell=1}^{m} R_{j,i}R_{\ell,i}\hat{\mathscr{C}}_{\epsilon,j,\ell}^{(n,N)}(t,t)}\right\}\right)$$

Estimating optimal routing matrix

Estimate:

$$\hat{R}_{\epsilon,\delta}^{(n,N)} \in \arg\min_{R \in \mathcal{R}} \left\{ \max_{i \in \{1,\dots,k\}} \left\{ \hat{P}_{\epsilon,\delta,i}^{(n,N)}(R) \right\} \right\}$$

Estimating optimal routing matrix

Estimate:

$$\hat{R}_{\epsilon,\delta}^{(n,N)} \in \arg\min_{R\in\mathcal{R}} \left\{ \max_{i\in\{1,\dots,k\}} \left\{ \hat{P}_{\epsilon,\delta,i}^{(n,N)}(R) \right\} \right\}$$

Theorem

For any finite set of routing matrices $\mathcal{R}$, we have

$$\lim_{n\to\infty} \lim_{N\to\infty} \frac{1}{n} \log \left(\frac{\max_{i\in\{1,\dots,k\}} \left\{ \mathbb{P}\left(Q_i^{(n)}\left(\hat{R}_{\epsilon,\delta}^{(n,N)} \right) > b_i \right) \right\}}{\min_{R\in\mathcal{R}} \left\{ \max_{i\in\{1,\dots,k\}} \left\{ \mathbb{P}\left(Q_i^{(n)}(R) > b_i \right) \right\} \right\}} \right) = 0, \quad a.s.,$$

for all ϵ and δ small enough.

Simulations

Simulations' setup

- $n = 1000$

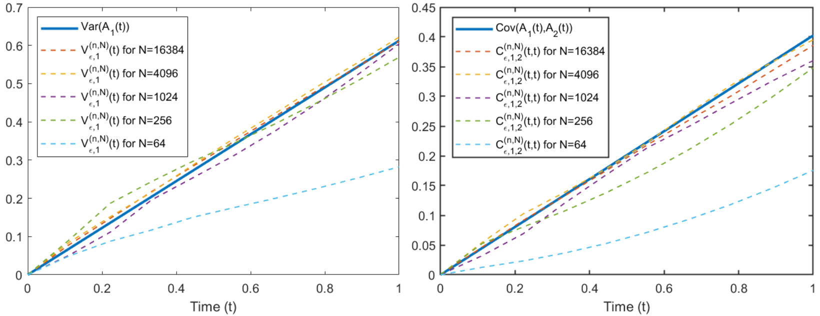
Simulations' setup

- $n = 1000$
- $m = k = 2$ and $R = Id$

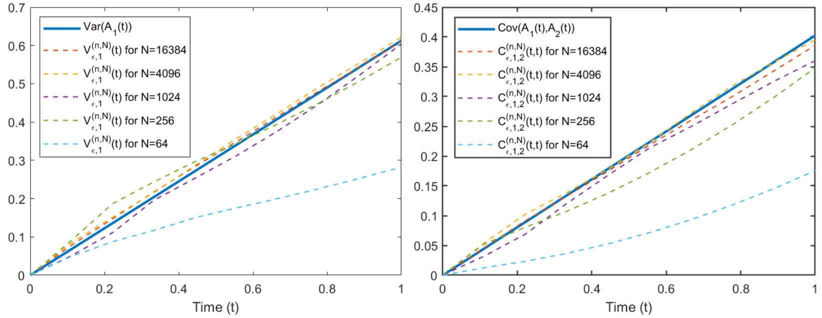
Simulations' setup

- $n = 1000$
- $m = k = 2$ and $R = Id$
- $\lambda_1 = \lambda_2 = 0.8$
- $\mu_1 = \mu_2 = 1$
- $\epsilon = 0.01$

Positively correlated Brownian motions

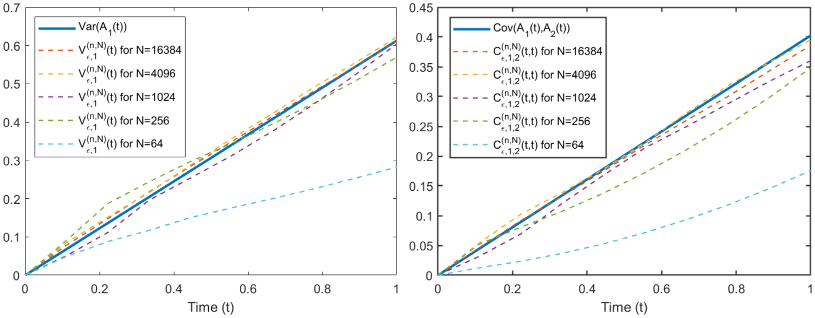


Positively correlated Brownian motions



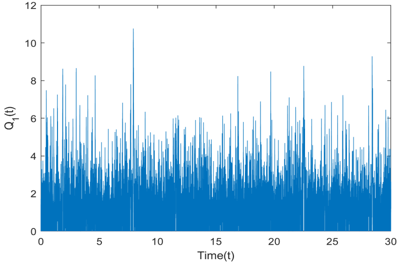
- Variance estimator works with $N \geq 256$ samples

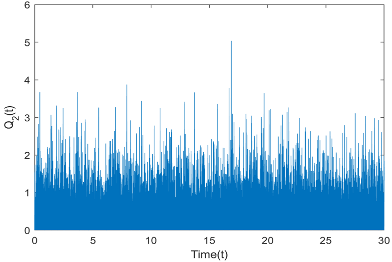
Positively correlated Brownian motions



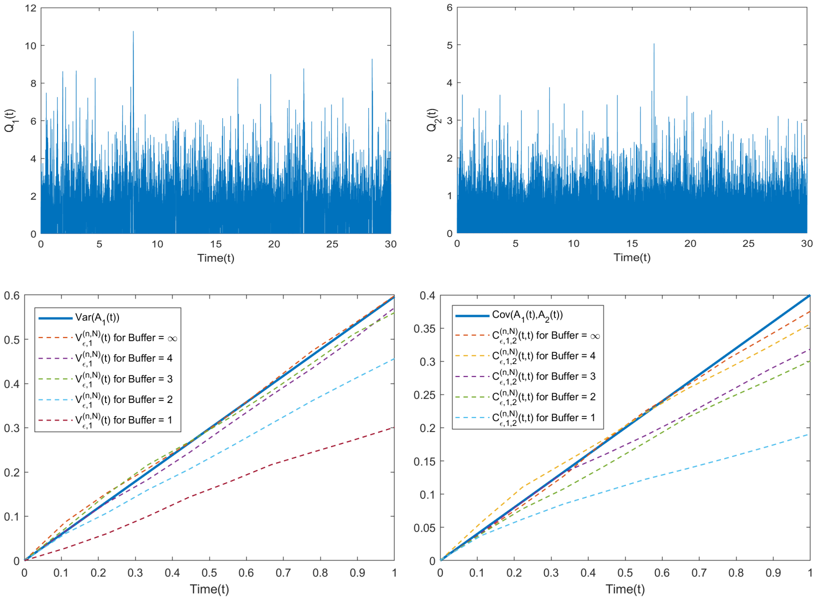
- Variance estimator works with $N \geq 256$ samples
- Covariance estimator works with $N \geq 1024$ samples

Positively correlated Brownian motions





Positively correlated Brownian motions



Positively correlated Brownian motions

Why does it work so well with finite buffers?

- Finite buffer truncates steady-state distribution

Positively correlated Brownian motions

Why does it work so well with finite buffers?

- Finite buffer truncates steady-state distribution

But:

- Large-deviation approxs do not depend on buffer size

Positively correlated Brownian motions

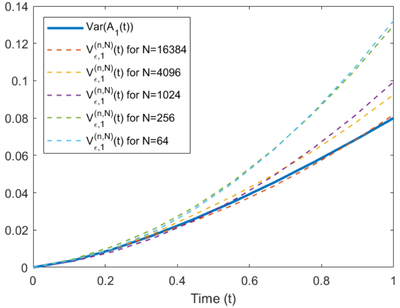
Why does it work so well with finite buffers?

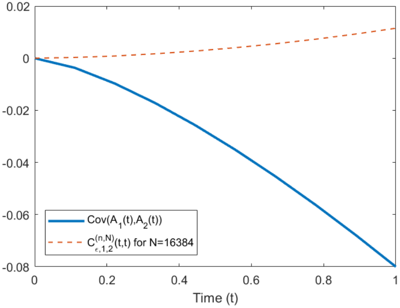
- Finite buffer truncates steady-state distribution

But:

- Large-deviation approxs do not depend on buffer size
- Estimates depend on an infimum, not whole distribution

Negatively correlated fractional Brownian motions





- Variance estimator works as expected
- Covariance estimator fails as expected

Conclusions

Conclusions

- Inputs' variances can be recovered from queue lengths
  $\Rightarrow$ Arrivals' covariances can be recovered with a few iterations

Conclusions

- Inputs' variances can be recovered from queue lengths $\Rightarrow$ Arrivals' covariances can be recovered with a few iterations

- Short-range dependent inputs + non-negative correlations $\Rightarrow$ Arrivals' covariances can be recovered from the queue lengths

Conclusions

- Inputs' variances can be recovered from queue lengths
  $\Rightarrow$ Arrivals' covariances can be recovered with a few iterations
- Short-range dependent inputs + non-negative correlations
  $\Rightarrow$ Arrivals' covariances can be recovered from the queue lengths
- Long-range dependent inputs + negative correlations
  $\Rightarrow$ Arrivals' covariances **cannot** be recovered from the queue lengths

Conclusions

- Inputs' variances can be recovered from queue lengths
  $\Rightarrow$ Arrivals' covariances can be recovered with a few iterations
- Short-range dependent inputs + non-negative correlations
  $\Rightarrow$ Arrivals' covariances can be recovered from the queue lengths
- Long-range dependent inputs + negative correlations
  $\Rightarrow$ Arrivals' covariances **cannot** be recovered from the queue lengths
- Can be extended to multi-path routing in acyclic networks (needs much more involved large-deviations results)

Thank you!