A Linear Algebraic Framework for Dynamic Scheduling Over Memory-Equipped Quantum Networks[1]

arXiv:2307.06009

Paolo Fittipaldi [A] Anastasios Giovanidis [A,B] Frédéric Grosshans[A]

[A]Sorbonne Université, CNRS, LIP6

[B]Ericsson AI Research & Systems

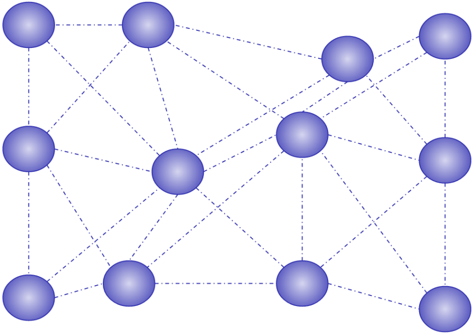
JFQI 2023

[1] P. F., Anastasios Giovanidis, and Frédéric Grosshans. “A Linear Algebraic Framework for Dynamic Scheduling Over Memory-Equipped Quantum Networks”. In: *IEEE Transactions on Quantum Engineering [Accepted for Publication]* (2023).

Outline

The Problem of Scheduling

Classical Networks

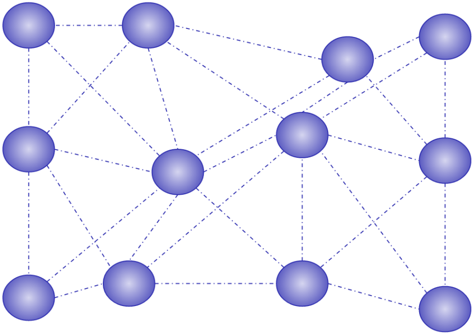


- Regulation of packet traffic along the network routes
- Standard solutions are well-known in the classical domain[2]

[2] L. Tassiulas and A. Ephremides. "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks". In: *IEEE Transactions on Automatic Control* 37.12 (1992), pp. 1936–1948.

The Problem of Scheduling

Quantum Networks



- Regulating which swapping operations happen at a given time
- Max Weight policies have recently been proven useful in quantum networks as well[3,4]

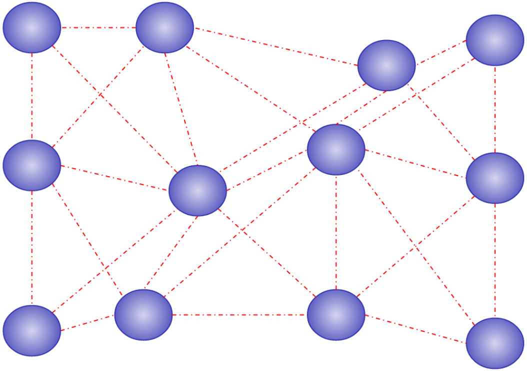
Thirupathaiah Vasantam and Don Towsley. "A throughput optimal scheduling policy for a quantum switch". In: *Quantum Computing, Communication, and Simulation II*. ed. by Philip R. Hemmer and Alan L. Migdall. Vol. 12015. International Society for Optics and Photonics. SPIE, 2022, p. 1201505.

P. F., Anastasios Giovanidis, and Frédéric Grosshans. "A Linear Algebraic Framework for Quantum Internet Dynamic Scheduling". In: *IEEE International Conference on Quantum Computing and Engineering (QCE22)*. Broomfield, CO, United States, Sept. 2022.

Contributions

- **Dynamically controlled scheduling framework for arbitrary topologies and multiple commodities**;
- Proposal of two classes of scheduling policies;
- Benchmarking of the proposed policies over nontrivial topologies;

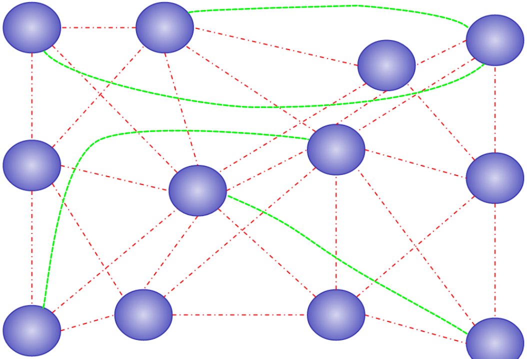
Notation



Arbitrary connected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$:

- Switches located at $\mathcal{V}$
- (Lossy) fiber links along $\mathcal{E}$
- Additionally, $\tilde{\mathcal{E}} = \mathcal{V} \times \mathcal{V}$ contains *all* possible edges

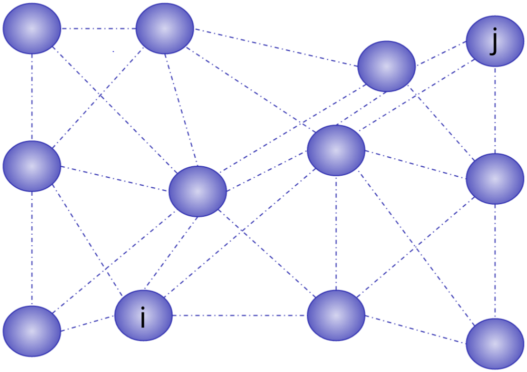
Notation



Arbitrary connected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$:

- Switches located at $\mathcal{V}$
- (Lossy) fiber links along $\mathcal{E}$
- Additionally, $\tilde{\mathcal{E}} = \mathcal{V} \times \mathcal{V}$ contains *all* possible edges

A Bird's-Eye View



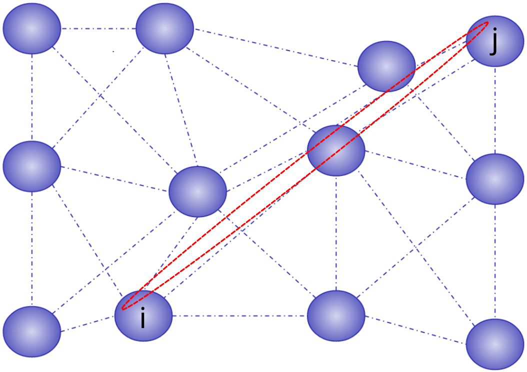
$\forall i, j \in \tilde{\mathcal{E}}$:

- An *Ebit Queue* $q_{ij}(t)$ keeps track of the stored entangled pairs
- A *Demand Queue* $d_{ij}(t)$ stores user demand (if any).

Objective

Linear equations that describe their intertwined evolution.

A Bird's-Eye View



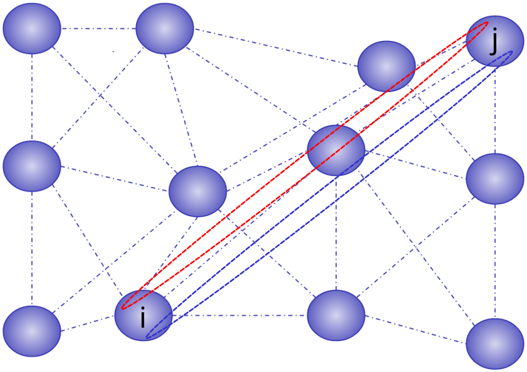
$\forall i, j \in \tilde{\mathcal{E}}$:

- An *Ebit Queue* $q_{ij}(t)$ keeps track of the stored entangled pairs
- A *Demand Queue* $d_{ij}(t)$ stores user demand (if any).

Objective

Linear equations that describe their intertwined evolution.

A Bird's-Eye View



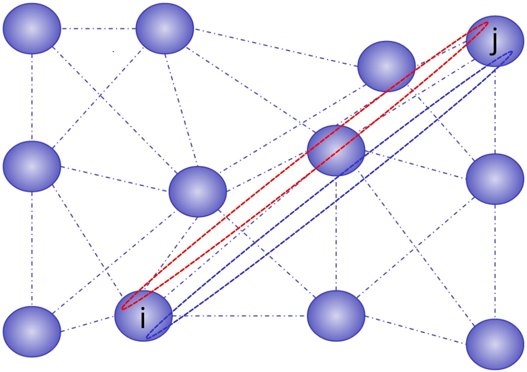
$\forall i, j \in \tilde{\mathcal{E}}$:

- An *Ebit Queue* $q_{ij}(t)$ keeps track of the stored entangled pairs
- A *Demand Queue* $d_{ij}(t)$ stores user demand (if any).

Objective

Linear equations that describe their intertwined evolution.

A Bird's-Eye View



$\forall i, j \in \tilde{\mathcal{E}}$:

- An *Ebit Queue* $q_{ij}(t)$ keeps track of the stored entangled pairs
- A *Demand Queue* $d_{ij}(t)$ stores user demand (if any).

Objective

Linear equations that describe their intertwined evolution.

Ebit Queues

$\forall i, j \in \tilde{\mathcal{E}}$:

$$\begin{bmatrix} q_{AB}(t+1) \\ q_{BC}(t+1) \\ q_{AC}(t+1) \\ \cdots \end{bmatrix} = \begin{bmatrix} q_{AB}(t) \\ q_{BC}(t) \\ q_{AC}(t) \\ \cdots \end{bmatrix} + \begin{bmatrix} a_{AB}(t) \\ a_{BC}(t) \\ a_{AC}(t) \\ \cdots \end{bmatrix} - \begin{bmatrix} \ell_{AB}(t) \\ \ell_{BC}(t) \\ \ell_{AC}(t) \\ \cdots \end{bmatrix} \pm \text{scheduling}$$

Ebit Queues

$\forall i, j \in \tilde{\mathcal{E}}$:

$$\begin{bmatrix} q_{AB}(t+1) \\ q_{BC}(t+1) \\ q_{AC}(t+1) \\ \dots \end{bmatrix} = \begin{bmatrix} q_{AB}(t) \\ q_{BC}(t) \\ q_{AC}(t) \\ \dots \end{bmatrix} + \begin{bmatrix} a_{AB}(t) \\ a_{BC}(t) \\ a_{AC}(t) \\ \dots \end{bmatrix} - \begin{bmatrix} \ell_{AB}(t) \\ \ell_{BC}(t) \\ \ell_{AC}(t) \\ \dots \end{bmatrix} \pm \text{scheduling}$$

Ebit Queues

Notation for Swapping

$$r_{i[j]k}(t) = n \in \mathbb{N}$$

Example

$r_{A[B]C}(2) = 3 \rightarrow$ Three swapping operations at node B from queues AB, BC to AC are scheduled to happen at time step 2.

Ebit Queues

Notation for Swapping

$$r_{i[j]k}(t) = n \in \mathbb{N}$$

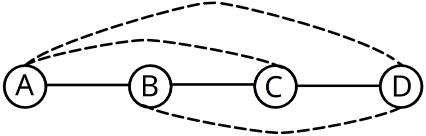
Example

$r_{A[B]C}(2) = 3 \rightarrow$ Three swapping operations at node B from queues AB, BC to AC are scheduled to happen at time step 2.

$$\boldsymbol{r}(t) = \begin{bmatrix} r_{A[B]C}(t) & r_{B[C]D}(t) & \dots \end{bmatrix}$$

Ebit Queues

The $\boldsymbol{M}$ Matrix



	$A[B]C$	$B[C]D$	$A[B]D$	$A[C]D$
AB	-1	0	-1	0
BC	-1	-1	0	0
CD	0	-1	0	-1
AC	$+1$	0	0	-1
BD	0	$+1$	-1	0
AD	0	0	$+1$	$+1$

$$\boldsymbol{q}(t+1) = \boldsymbol{q}(t) - \boldsymbol{\ell}(t) + \boldsymbol{a}(t) + \boldsymbol{M}\boldsymbol{r}(t)$$

Consuming Pairs to Serve Demands

$$\tilde{\boldsymbol{M}} = \begin{bmatrix} \boldsymbol{M} & | & -\mathbb{I}_{N_{\text{queues}}} \end{bmatrix}$$

	$A[B]C$	$B[C]D$	$A[B]D$	$A[C]D$	AB	BC	CD	AC	BD	AD
AB	−1	0	−1	0	−1	0	0	0	0	0
BC	−1	−1	0	0	0	−1	0	0	0	0
CD	0	−1	0	−1	0	0	−1	0	0	0
AC	+1	0	0	−1	0	0	0	−1	0	0
BD	0	+1	−1	0	0	0	0	0	−1	0
AD	0	0	+1	+1	0	0	0	0	0	−1

$$\boldsymbol{r}(t) = \begin{bmatrix} r_{A[B]C}(t) & r_{B[C]D}(t) & \dots & | & r_{AB}(t) & r_{BC}(t) & \dots \end{bmatrix}$$

Demand Queues

$$\boldsymbol{q}(t+1) = \boldsymbol{q}(t) + \boldsymbol{a}(t) - \boldsymbol{\ell}(t) + \tilde{\boldsymbol{M}}\boldsymbol{r}(t)$$

- $$\boldsymbol{d}(t+1) = \boldsymbol{d}(t) + \boldsymbol{b}(t) \qquad + \tilde{\boldsymbol{N}}\boldsymbol{r}(t)$$
- $$\tilde{\boldsymbol{N}} = \left[\mathbb{O}_{N_{\text{queues}} \times N_{\text{transitions}}} \quad | \quad -\mathbb{I}_{N_{\text{queues}}}\right]$$

Recap

- $\forall (i,j) \in \tilde{\mathcal{E}}$ there is an ebit queue and a demand queue.
- Scheduling $\rightarrow$ Deciding $\boldsymbol{r}(t)$ given some information about $\boldsymbol{q}$ and $\boldsymbol{d}$
- Information about routing and topology is encoded in the $\boldsymbol{M}$ matrix.

Deriving a Scheduling Policy

The Lyapunov Function

$$V(t) = \frac{1}{2}\boldsymbol{d}^T(t)\boldsymbol{d}(t) = d_{AB}^2 + d_{BC}^2 + d_{CD}^2 \cdots$$

- Scalar measure of the overall system stability;
- Standard tool in classical Network Science;
- Akin to a potential in physics;

Deriving a Scheduling Policy

Lyapunov Drift

$$V(t) = \frac{1}{2}\boldsymbol{d}^T(t)\boldsymbol{d}(t)$$

$$\Delta V = \frac{1}{2}\mathbb{E}[\boldsymbol{d}^T(t+1)\boldsymbol{d}(t+1) - \boldsymbol{d}^T(t)\boldsymbol{d}(t)\big|\mathcal{I}(t)]$$

Expanding the mathematics, we obtain a **controllable expression** that gauges the effects of a given scheduling decision on the system.

Deriving a Scheduling Policy

The Full Information Quadratic Scheduler

$$\Delta V = \frac{1}{2}\mathbb{E}[\boldsymbol{d}^T(t+1)\boldsymbol{d}(t+1) - \boldsymbol{d}^T(t)\boldsymbol{d}(t)\big|\mathcal{I}(t)] \rightarrow \begin{cases} \boldsymbol{r}(t) = \arg\min\Big(\boldsymbol{w}^T(t)\cdot\boldsymbol{r}(t) + \\ \qquad +\frac{1}{2}\boldsymbol{r}(t)^T\tilde{\boldsymbol{N}}^T\tilde{\boldsymbol{N}}\boldsymbol{r}(t)\Big) \\ \text{s.t. } \boldsymbol{r}(t) \in \mathcal{R}(t) \\ \boldsymbol{w}(t) = (\boldsymbol{d}(t) + \boldsymbol{b}(t))^T\tilde{\boldsymbol{N}} \end{cases}$$

$$\mathcal{R}(t) = \left\{\boldsymbol{r}(t) \in \mathbb{N}^d \;\middle|\; \begin{array}{l} -\tilde{\boldsymbol{M}}\boldsymbol{r}(t) \leq \boldsymbol{q}(t) - \boldsymbol{\ell}(t) + \boldsymbol{a}(t) \\ -\tilde{\boldsymbol{N}}\boldsymbol{r}(t) \leq \boldsymbol{d}(t) \qquad\qquad + \boldsymbol{b}(t) \end{array}\right\}$$

- At every time step, solve the optimization problem

\+ High performance

– Large communication requirements

– High computational cost

Deriving a Scheduling Policy

The Full Information Quadratic Scheduler

$$\Delta V = \frac{1}{2}\mathbb{E}[\boldsymbol{d}^T(t+1)\boldsymbol{d}(t+1) - \boldsymbol{d}^T(t)\boldsymbol{d}(t)|\mathcal{I}(t)] \rightarrow \begin{cases} \boldsymbol{r}(t) = \arg\min\Big(\boldsymbol{w}^T(t)\cdot\boldsymbol{r}(t) + \\ \qquad + \frac{1}{2}\boldsymbol{r}(t)^T\tilde{\boldsymbol{N}}^T\tilde{\boldsymbol{N}}\boldsymbol{r}(t)\Big) \\ \text{s.t. } \boldsymbol{r}(t) \in \mathcal{R}(t) \\ \boldsymbol{w}(t) = (\boldsymbol{d}(t) + \boldsymbol{b}(t))^T\tilde{\boldsymbol{N}} \end{cases}$$

$$\mathcal{R}(t) = \left\{\boldsymbol{r}(t) \in \mathbb{N}^d \;\middle|\; \begin{matrix} -\tilde{\boldsymbol{M}}\boldsymbol{r}(t) \le \boldsymbol{q}(t) - \boldsymbol{\ell}(t) + \boldsymbol{a}(t) \\ -\tilde{\boldsymbol{N}}\boldsymbol{r}(t) \le \boldsymbol{d}(t) \qquad\qquad + \boldsymbol{b}(t) \end{matrix}\right\}$$

- At every time step, solve the optimization problem
- \+ High performance
- – Large communication requirements
- – High computational cost

Deriving a Scheduling Policy

The Full Information Quadratic Scheduler

$$\Delta V = \frac{1}{2}\mathbb{E}[\boldsymbol{d}^T(t+1)\boldsymbol{d}(t+1) - \boldsymbol{d}^T(t)\boldsymbol{d}(t) \big| \mathcal{I}(t)] \rightarrow \begin{cases} \boldsymbol{r}(t) = \arg\min \Big(\boldsymbol{w}^T(t) \cdot \boldsymbol{r}(t) + \\ \qquad + \frac{1}{2}\boldsymbol{r}(t)^T \tilde{\boldsymbol{N}}^T \tilde{\boldsymbol{N}} \boldsymbol{r}(t) \Big) \\ \text{s.t. } \ \boldsymbol{r}(t) \in \mathcal{R}(t) \\ \boldsymbol{w}(t) = (\boldsymbol{d}(t) + \boldsymbol{b}(t))^T \tilde{\boldsymbol{N}} \end{cases}$$

$$\mathcal{R}(t) = \left\{ \boldsymbol{r}(t) \in \mathbb{N}^d \ \middle| \ \begin{array}{ll} -\tilde{\boldsymbol{M}}\boldsymbol{r}(t) \leq \boldsymbol{q}(t) - \boldsymbol{\ell}(t) + \boldsymbol{a}(t) \\ -\tilde{\boldsymbol{N}}\boldsymbol{r}(t) \leq \boldsymbol{d}(t) \qquad\qquad + \boldsymbol{b}(t) \end{array} \right\}$$

- At every time step, solve the optimization problem
- \+ High performance
- – Large communication requirements
- – High computational cost

Deriving a Scheduling Policy

The Full Information Quadratic Scheduler

$$\Delta V = \frac{1}{2}\mathbb{E}[\boldsymbol{d}^T(t+1)\boldsymbol{d}(t+1) - \boldsymbol{d}^T(t)\boldsymbol{d}(t) \big| \mathcal{I}(t)] \rightarrow \begin{cases} \boldsymbol{r}(t) = \arg\min \Big(\boldsymbol{w}^T(t) \cdot \boldsymbol{r}(t) + \\ \qquad + \frac{1}{2}\boldsymbol{r}(t)^T \tilde{\boldsymbol{N}}^T \tilde{\boldsymbol{N}} \boldsymbol{r}(t) \Big) \\ \text{s.t. } \ \boldsymbol{r}(t) \in \mathcal{R}(t) \\ \boldsymbol{w}(t) = (\boldsymbol{d}(t) + \boldsymbol{b}(t))^T \tilde{\boldsymbol{N}} \end{cases}$$

$$\mathcal{R}(t) = \left\{ \boldsymbol{r}(t) \in \mathbb{N}^d \ \middle| \ \begin{array}{l} -\tilde{\boldsymbol{M}}\boldsymbol{r}(t) \leq \boldsymbol{q}(t) - \boldsymbol{\ell}(t) + \boldsymbol{a}(t) \\ -\tilde{\boldsymbol{N}}\boldsymbol{r}(t) \leq \boldsymbol{d}(t) \qquad\qquad + \boldsymbol{b}(t) \end{array} \right\}$$

- At every time step, solve the optimization problem
- \+ High performance
- – Large communication requirements
- – High computational cost

Alternative Scheduling Policies

PROBLEMS

- Quadratic integer optimization is computationally expensive;
- Full-Information scheduling has impossible communication constraints;

SOLUTIONS

- Suppress the quadratic term in the optimization problem $\longrightarrow$ **MAX-WEIGHT SCHEDULING**;
- Replace all the information we don't have with educated guesses $\longrightarrow$ **LOCAL INFORMATION SCHEDULING**

Alternative Scheduling Policies

PROBLEMS

- Quadratic integer optimization is computationally expensive;
- Full-Information scheduling has impossible communication constraints;

SOLUTIONS

- Suppress the quadratic term in the optimization problem $\longrightarrow$ **MAX-WEIGHT SCHEDULING**;
- Replace all the information we don't have with educated guesses $\longrightarrow$ **LOCAL INFORMATION SCHEDULING**

Alternative Scheduling Policies

PROBLEMS

- Quadratic integer optimization is computationally expensive;
- Full-Information scheduling has impossible communication constraints;

SOLUTIONS

- Suppress the quadratic term in the optimization problem → **MAX-WEIGHT SCHEDULING**;
- Replace all the information we don't have with educated guesses → **LOCAL INFORMATION SCHEDULING**

Alternative Scheduling Policies

PROBLEMS

- Quadratic integer optimization is computationally expensive;
- Full-Information scheduling has impossible communication constraints;

SOLUTIONS

- Suppress the quadratic term in the optimization problem → **MAX-WEIGHT SCHEDULING**;
- Replace all the information we don't have with educated guesses → **LOCAL INFORMATION SCHEDULING**

Alternative Scheduling Policies

PROBLEMS

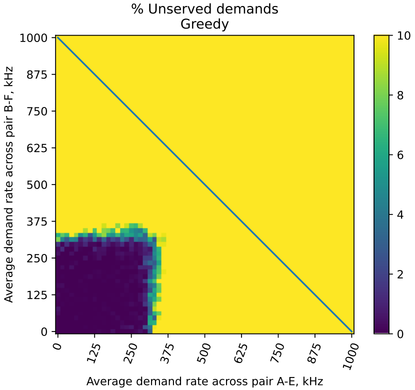
- Quadratic integer optimization is computationally expensive;
- Full-Information scheduling has impossible communication constraints;

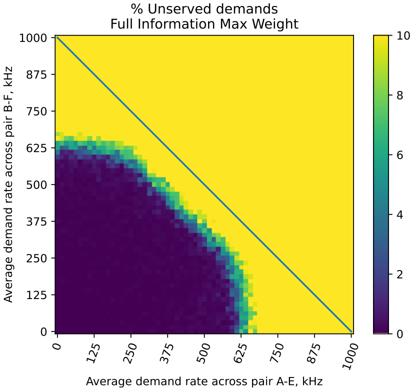
SOLUTIONS

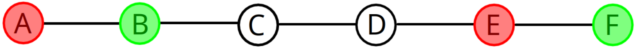
- Suppress the quadratic term in the optimization problem $\longrightarrow$ **MAX-WEIGHT SCHEDULING**;
- Replace all the information we don't have with educated guesses $\longrightarrow$ **LOCAL INFORMATION SCHEDULING**

Result Example

FIMW Scheduler, Simple Topology

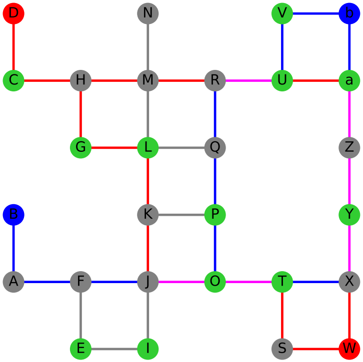


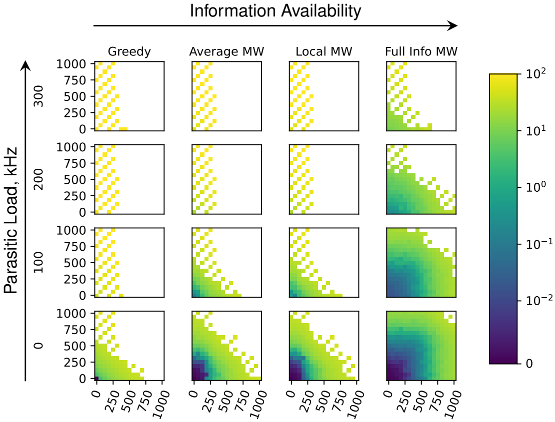




Results

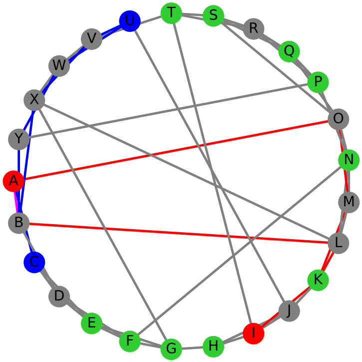
Grid with Holes

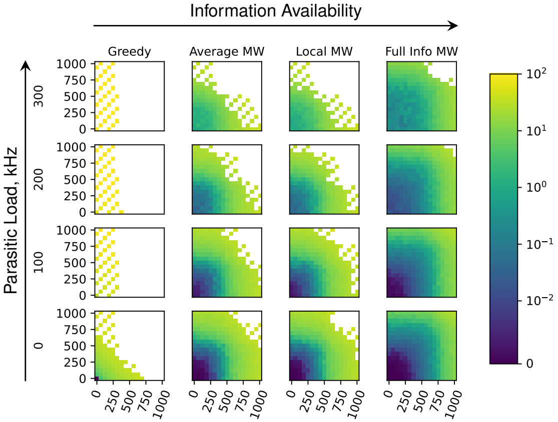




Results

Watts-Strogatz Graph





Summary

- We introduced a multicommodity dynamically controlled framework for scheduling on general topologies;
- We derived a class of scheduling policies for quantum networks that minimizes the square norm of the demand backlog;
- Our framework was demonstrated to be useful as a benchmarking tool for arbitrary policies over non-trivial topologies.
- Outlook
  - Introducing quantum noise in the framework;
  - Provide analytical demonstration of stability and optimality of policies.

References

- **F., P.**, Anastasios Giovanidis, and Frédéric Grosshans. "A Linear Algebraic Framework for Dynamic Scheduling Over Memory-Equipped Quantum Networks". In: *IEEE Transactions on Quantum Engineering* ***[Accepted for Publication]*** (2023).
- **F., P.**, Anastasios Giovanidis, and Frédéric Grosshans. "A Linear Algebraic Framework for Quantum Internet Dynamic Scheduling". In: *IEEE International Conference on Quantum Computing and Engineering (QCE22)*. Broomfield, CO, United States, Sept. 2022.
- Tassiulas, L. and A. Ephremides. "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks". In: *IEEE Transactions on Automatic Control* 37.12 (1992), pp. 1936–1948.
- Vasantam, Thirupathaiah and Don Towsley. "A throughput optimal scheduling policy for a quantum switch". In: *Quantum Computing, Communication, and Simulation II*. Ed. by Philip R. Hemmer and Alan L. Migdall. Vol. 12015. International Society for Optics and Photonics. SPIE, 2022, p. 1201505.