Stability of Decentralized Queueing Networks —Beyond Complete Bipartite Cases

Hu Fu
Shanghai University of Finance and Economics (SHUFE)

Joint work with Qun Hu (SHUFE) and Jia'nan Lin (RPI)

Centralized vs. Decentralized Systems

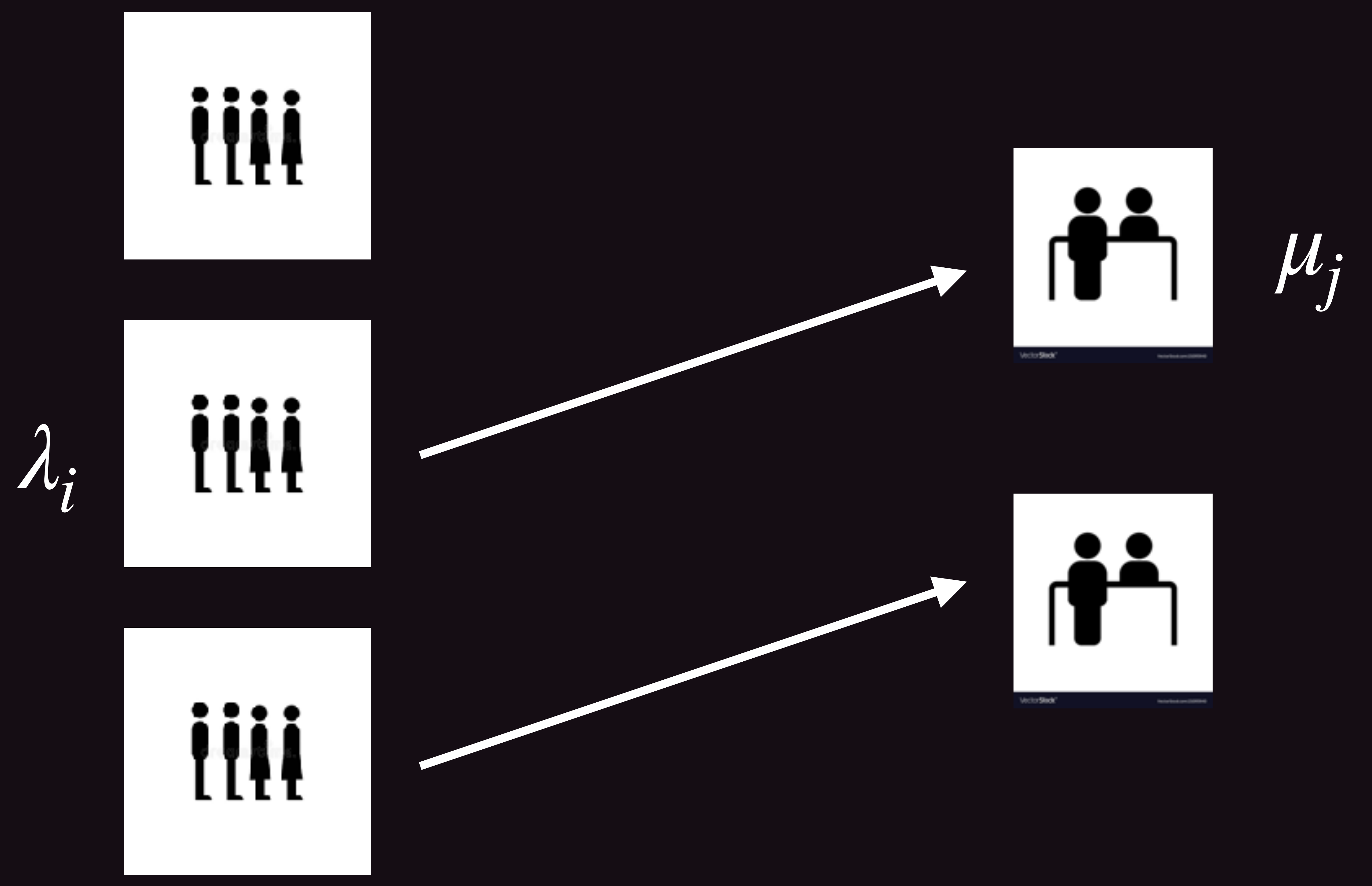
- Price of Anarchy [Koutsoupias & Papadimitriou 99]
- Among many examples:
  - Routing in congestion games [Roughgarden & Tardos 02..]
  - Auctions [Christodoulou, Kovács, Schapira 08]
- Gaitonde & Tardos, 20: Queueing systems
  - Game of many rounds
  - Outcomes of each round affect future rounds

Queueing System of Gaitonde & Tardos

n queues, m servers

At each time step:

- A customer joins queue i w.p. λ_i
- Each queue chooses a server and sends a customer (of earliest arrival)
- Each server picks one of the customers sent to it
- Server j successfully serves its customer w.p. μ_j

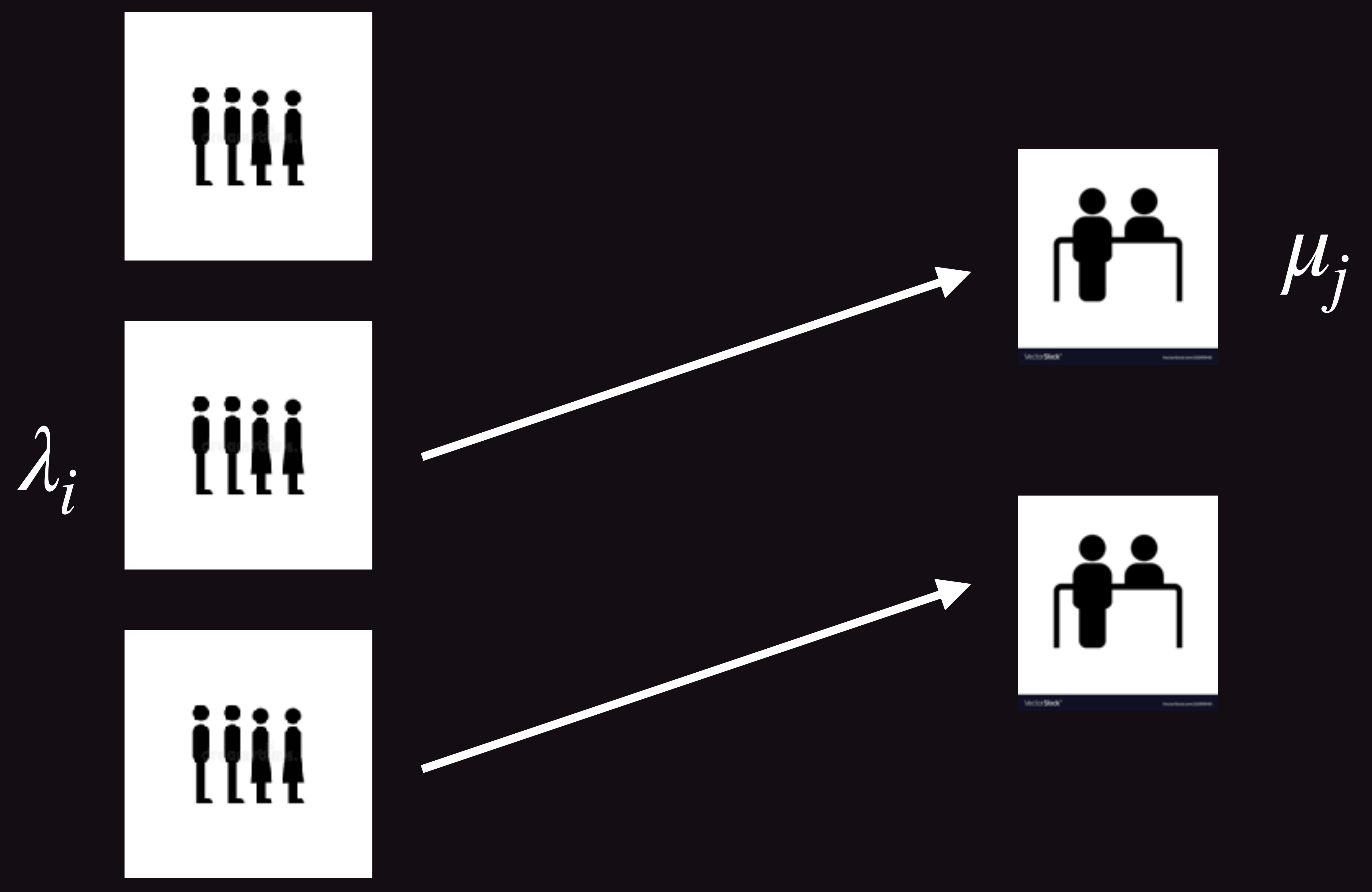


Queueing System of Gaitonde & Tardos

n queues, m servers

At each time step:

- A customer joins queue i w.p. λ_i
- Each queue chooses a server and sends a customer
- Each server picks one of the customers sent to it
- Server j successfully serves its customer w.p. μ_j



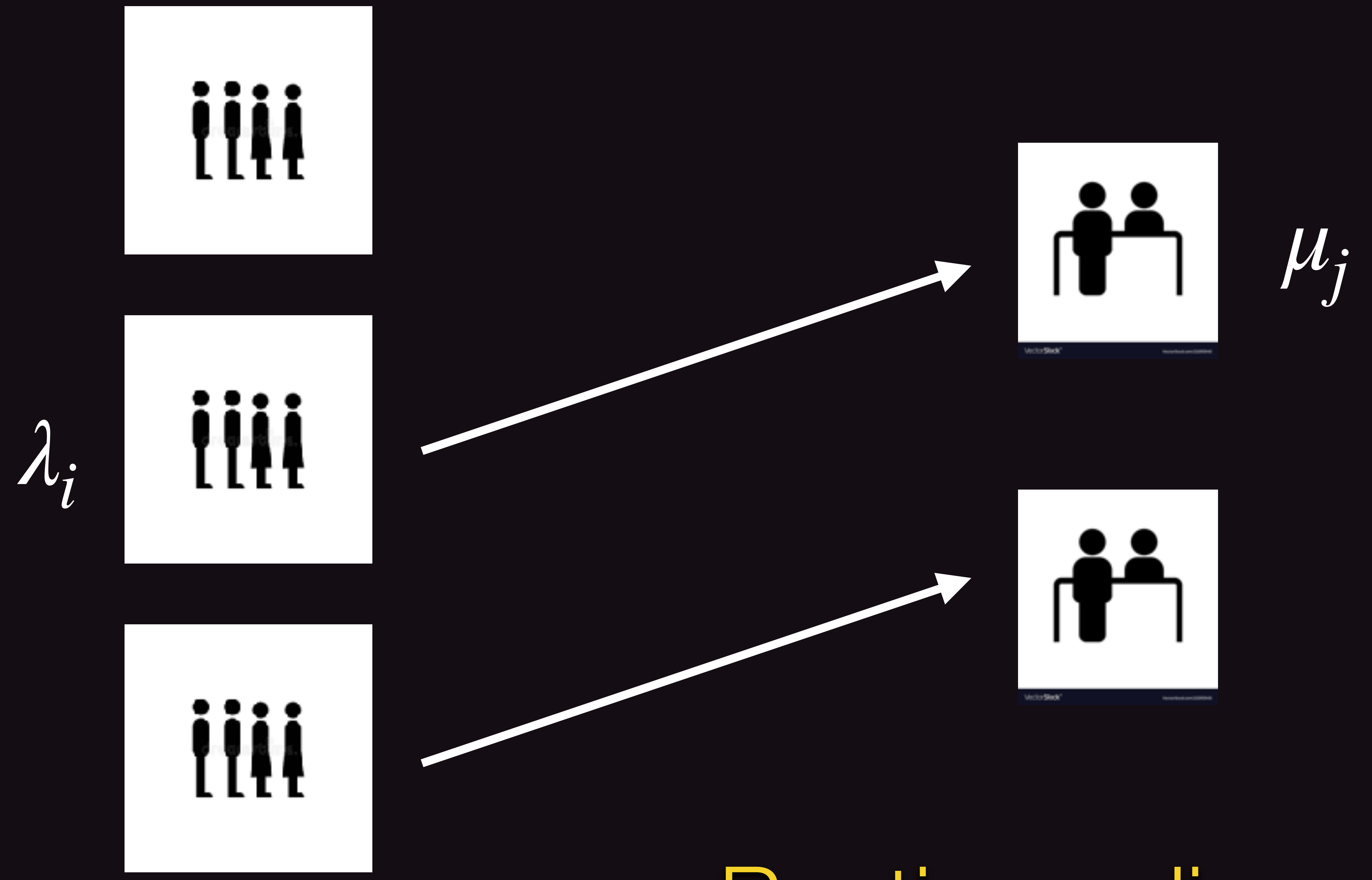
Service priority

Queueing System of Gaitonde & Tardos

n queues, m servers

At each time step:

- A customer joins queue i w.p. λ_i
- Each queue chooses a server and sends a customer (of earliest arrival)
- Each server picks one of the customers sent to it
- Server j successfully serves its customer w.p. μ_j



Routing policy

System Desideratum: Stability

- Roughly put, we'd like the queue lengths not to explode
- More precisely, write Q_t^i as the number of customers in queue i after time t
  - $Q_t := \sum_i Q_t^i$
  - (Strongly) stable: $\forall \alpha > 0, \mathbb{E}[(Q_t)^\alpha] = O_\alpha(1)$.

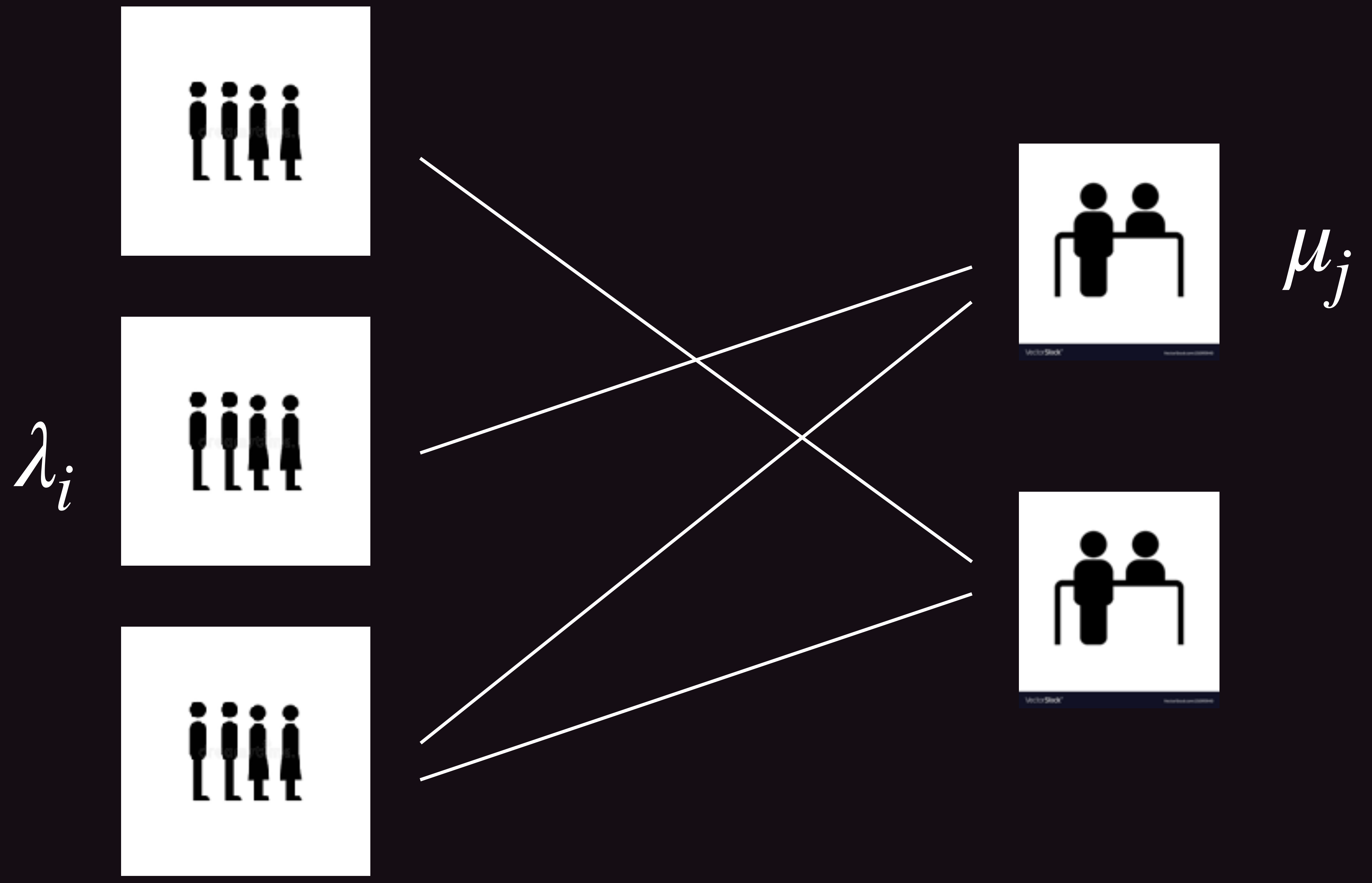
Results in a Nutshell (Part 1)

- Gaitonde & Tardos [EC 20]:
  - Characterization of systems that can be made stable under a centralized policy
  - Sufficient condition for systems that are stable as long as each queue uses a no-regret learning strategy
- This work:
  - Generalized both results when not all servers can serve all queues
    - Our conditions are similar to Gaitonde & Tardos's, and include theirs as a special case.
  - Similar results do *not* extend when the network has multiple layers
    - We give modifications of the service priority and the queues' utilities that restore comparable results.

Results in a Nutshell (Part 2)

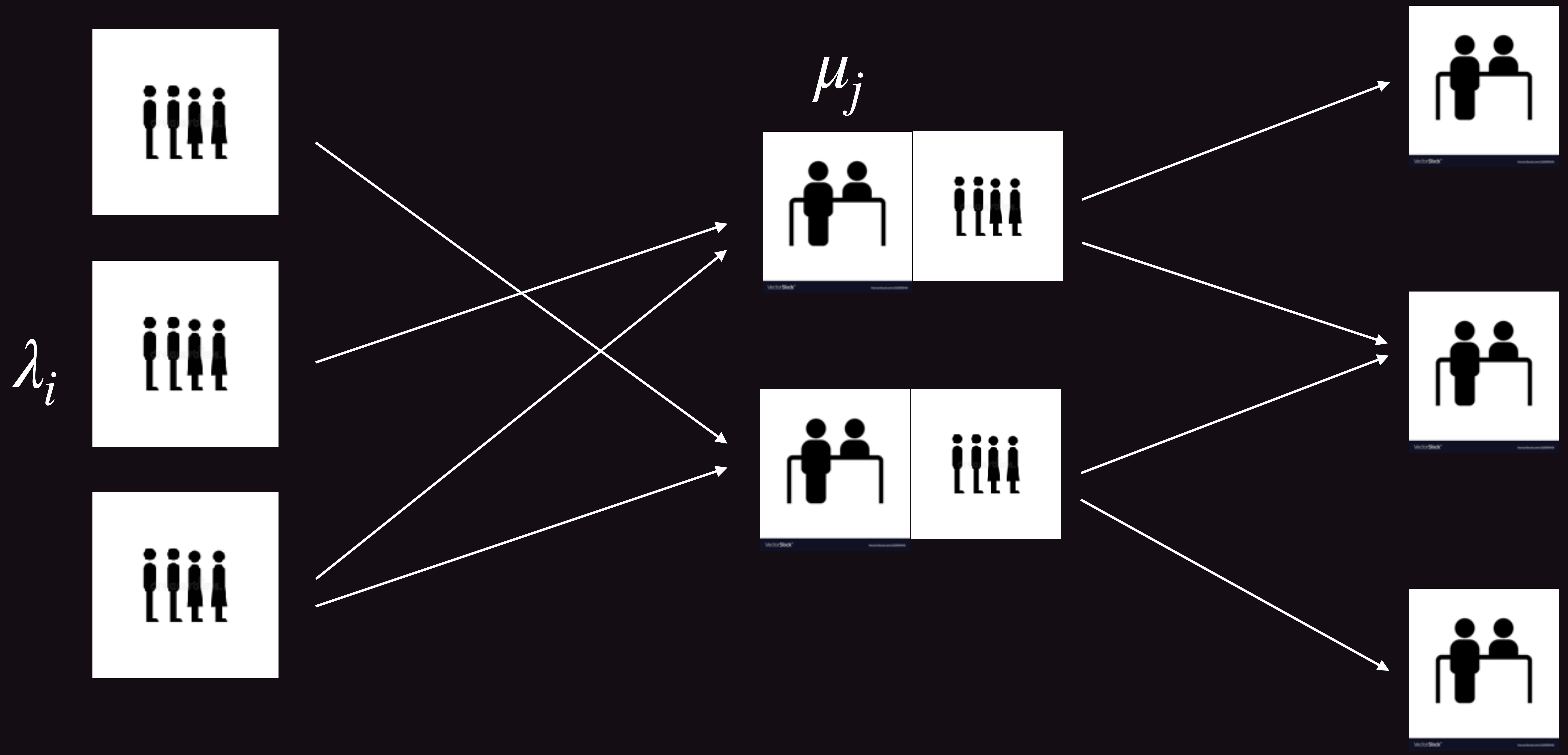
- Gaitonde & Tardos [EC 21]:
  - What if the queues do not alter their strategy from step to step, but sample a server from a fixed distribution? Equilibria can be defined in terms of these distributions.
  - Conditions guaranteeing a system to be stable under any such equilibrium.
- This work:
  - Generalized such conditions (not as tight) when not all servers can serve all queues

Queueing System with Incomplete Bipartite Graphs



Server j can process a packet from queue i if (i, j) is an edge

Queueing System on a DAG



After a node successfully processes a packet, the packet joins the queue at the node for the next stage.

Stability under Centralized Policy

- It never benefits a central planner to send two packets to the same server
  - In the bipartite case, a central planner picks a matching at each step
  - The matching may be sampled from a distribution
- Can this distribution be history independent?
  - It turns out that this is so.

Stability Conditions for Centralized Systems

Thm [F., Hu, Lin] A queueing system on a bipartite graph of n queues and m servers with arrival rates $\lambda = (\lambda_1, \cdots, \lambda_n)$ and processing rates $\mu = (\mu_1, \ldots, \mu_m)$ can be stable under a centralized policy if and only if there is a fractional matching matrix $P \in [0,1]^{n\times m}$ with $P\mu \succ \lambda$.

element-wise >

Recall: $P \in [0,1]^{n\times m}$ is a fractional matching matrix if $\sum_{i:(i,j)\in E} P_{ij} \leq 1, \forall j \in [m]$, and

$$\sum_{j:(i,j)\in E} P_{ij} \leq 1, \forall i \in [n].$$

Stability Conditions for Centralized Systems

Thm [F., Hu, Lin] A queueing system on a bipartite graph of n queues and m servers with arrival rates $\lambda = (\lambda_1, \cdots, \lambda_n)$ and processing rates $\mu = (\mu_1, \ldots, \mu_m)$ can be stable under a centralized policy if and only if there is a fractional matching matrix $P \in [0,1]^{n \times m}$ with $P\mu \succ \lambda$.

If the bipartite graph is complete, P is doubly stochastic; this condition requires μ to majorize λ. This is indeed the condition given by Gaitonde & Tardos.

Stability Conditions for Centralized Systems

Thm [F., Hu, Lin] A queueing system on a DAG $G = (V, E)$ can be stable under a centralized policy if and only if there exists $\mathbf{z} \in [0,1]^E$ such that

$$\lambda_i < \sum_{j:(i,j)\in E} z_{ij}\mu_j$$ for all first layer queue i

$$\mu_i \sum_{j:(j,i)\in E} z_{ji} < \sum_{j:(i,j)\in E} z_{ij}\mu_j$$ for all middle layer server i

$$\sum_{j:(j,i)\in E} z_{ji} \le 1, \sum_{j:(i,j)\in E} z_{ij} \le 1$$ for all node i

View z_{ij} as Pr[i chooses j]
at each time step

Impatient Utilities

- Let $a_i(t)$ be the server chosen by queue i at time step t
- Let $u_t^i(a_i(t), \mathbf{a}_{-i}(t) \mid \mathscr{F}_t)$ be the utility of queue i at time step t
  - $\mathscr{F}_t$ is the history up to time t
- In the “impatient” model, Gaitonde & Tardos defined u_t^i as 1 if a packet from queue i is cleared during time step t, and 0 otherwise.

No-Regret Strategies

The regret of queue i after w steps is

$$\mathrm{Reg}_i(w) := \max_{j:(i,j)\in E} \sum_{t=1}^{w} u_t^i(j, \mathbf{a}_{-i}(t) \mid \mathscr{F}_t) - \sum_{t=1}^{w} u_t^i(a_i(t), \mathbf{a}_{-i}(t) \mid \mathscr{F}_t)$$

the actual histories!

best utility in hindsight by choosing a fixed server at each step

the actual cumulative utility of queue i

A routing policy is no regret if, for fixed $\delta \in (0,1)$, $\mathrm{Reg}_i(w) = o_\delta(w)$ w.p. $1-\delta$

No-regret strategies are well known to exist, e.g. MWU

Decentralized Stability in Complete Bipartite Graphs

Thm (Gaitonde & Tardos 20) If the following condition is satisfied, a queueing system on a bipartite graph is stable as long as all queues play no-regret strategies:

there is $\eta > 0$ such that $\frac{1}{2}(1-\eta)\mu$ majorizes λ.

Therefore, by doubling the processing capacities, one can guarantee that a centralized stable system is also stable with decentralized queues using no-regret strategies. The factor 2 is tight.

Dual Form of Centralized Stability Conditions

Lemma A queueing system on a bipartite graph of n queues and m servers with arrival rates $\lambda = (\lambda_1, \cdots, \lambda_n)$ and processing rates $\mu = (\mu_1, \ldots, \mu_m)$ can be stable under a centralized policy if and only if for any $\alpha \in \mathbb{R}^n_+$, there is a matching matrix $M \in \{0,1\}^{n \times m}$, such that $\alpha^\top M\mu > \alpha^\top \lambda$.

This is simply the dual form of the conditions we gave before, obtained via Farkas' lemma.

Stability under Decentralized No-Regret Policies

Thm (F., Hu, Lin) If the following condition is satisfied, a queueing system on a bipartite graph is stable as long as all queues play no-regret strategies:

(*) there exists $\eta > 0$, such that for any $\alpha \in \{0,1\}^n$, there is a matching matrix $M \in \{0,1\}^{n\times m}$, such that $\frac{1}{2}(1-\eta)\alpha^\top M\mu > \alpha^\top \lambda$.

Compare with the centralized condition: for any $\alpha \in \mathbb{R}_+^n$, there is a matching matrix $M \in \{0,1\}^{n\times m}$, such that $\alpha^\top M\mu > \alpha^\top \lambda$.

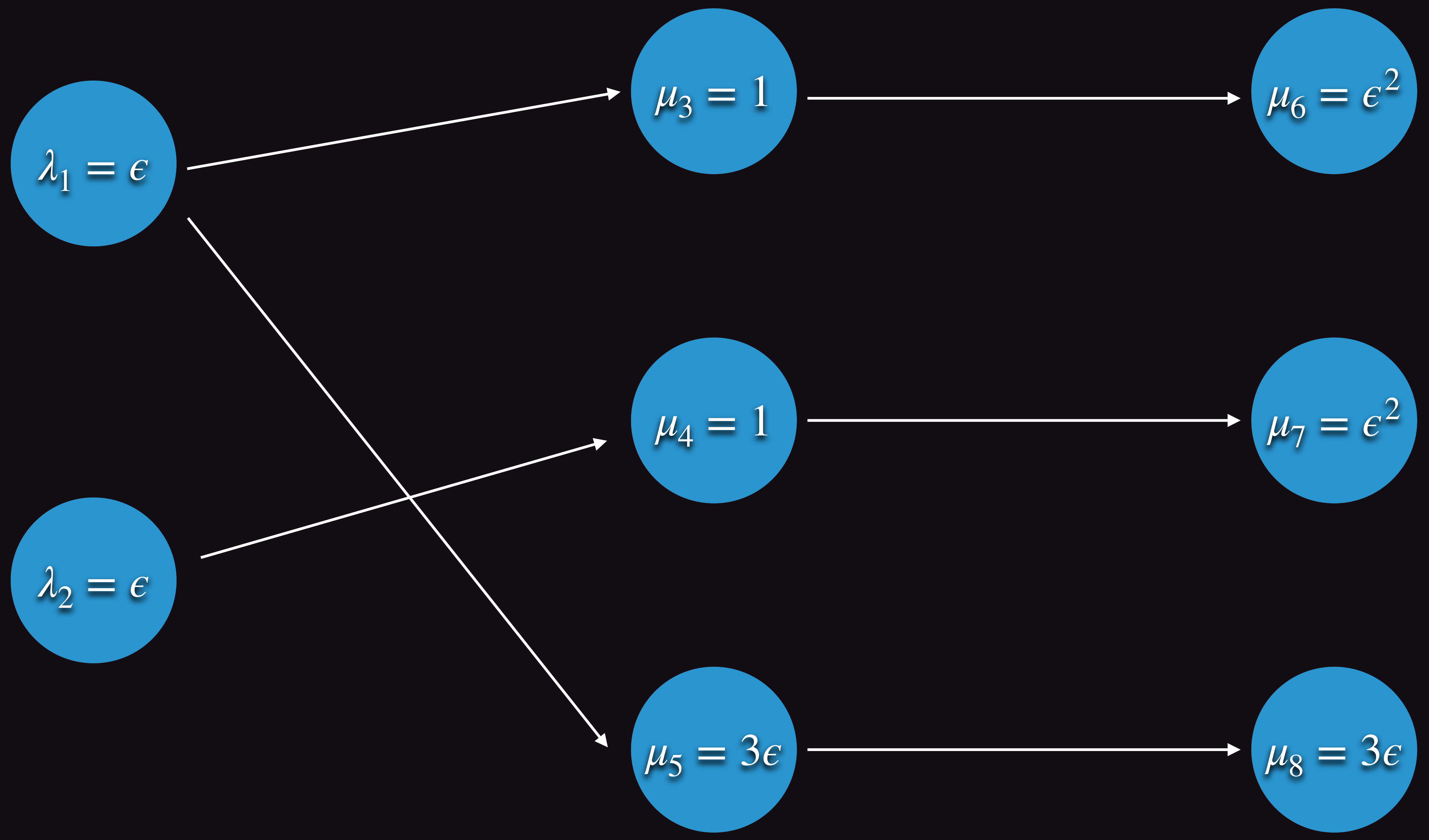
Stability under Decentralized No-Regret Policies

Thm If the following condition is satisfied, a queueing system on a bipartite graph is stable as long as all queues play no-regret strategies:

(*) there exists $\eta > 0$, such that for any $\alpha \in \{0,1\}^n$, there is a matching matrix $M \in \{0,1\}^{n \times m}$, such that $\frac{1}{2}(1-\eta)\alpha^\top M\mu > \alpha^\top\lambda$.

For complete bipartite graphs, the dual form of the centralized stability condition in fact only needs $\alpha \in \{0,1\}^n$. But this is with loss in general bipartite graphs.

Myopic Queues Fail in Multi-Layer Systems



This system is stable under a central policy

For stability under no-regret policies, the processing rate needs to increase by a factor of $\Omega(1/\epsilon)$.

New Utility and Service Priority

- Queues and servers should not do global calculation — otherwise why not implement some centralized policy?
- Goal: Use local information to overcome the myopia
- New utility: at time t, if queue i sends a packet to server j and has it successfully processed, queue i gains utility $Q_t^i - Q_t^j$.
- New service priority: pick the packet from the longest queue

Dual Form of Centralized Stability Conditions in DAG

Def. A path ensemble in a graph is a set of vertex-disjoint paths.

Lemma A queueing system on a DAG $G = (V, E)$ is stable under some centralized policy if and only if for any $\boldsymbol{\alpha} \in \mathbb{R}_+^V$, there is a path ensemble U, such that $\sum_{v \in S} \alpha_v \lambda_v < \sum_{(u,v) \in U} (\alpha_u - \alpha_v) \mu_j$.

nodes with no incoming edges

Consequence of Farkas' lemma

Stability under Decentralized No-Regret Policies

Thm (F., Hu, Lin) With the queue-length aware utilities and service priority, if the following condition is satisfied, a queueing system on a DAG is stable as long as all queues play no-regret strategies:

(*) there exists $\eta > 0$, such that for for any $\boldsymbol{\alpha} \in \mathbb{R}_+^V$, there is a path ensemble U, such that $\sum_{v \in S} \alpha_v \lambda_v < \frac{1}{2}(1-\eta) \sum_{(u,v) \in U} (\alpha_u - \alpha_v)\mu_j$.

“Patient” Queues

- What if the queues don’t adjust their strategy from step to step, but fix on one and observe their performance over long periods?
- Such a strategy is simply a distribution over the servers it can reach
- Let T_t^i be the age of the oldest packet in queue i at time t
- The utility of a queue is $\lim_{t\to\infty} \frac{T_t^i}{t}$ [Gaitonde & Tardos 21]
- One can then define Nash equilibria in this game and study their stability

Stability of Equilibria with Patient Queues

Thm (Gaitonde & Tardos 21) A queueing system on a complete bipartite graph is stable under all Nash equilibria if $(1 - \frac{1}{e})\mu$ strictly majorizes λ, and the factor $1 - \frac{1}{e}$ is tight.

Thm (F., Hu, Lin) A queueing system on a bipartite graph is stable under all Nash equilibria if there is $\eta > 0$ and a fractional matching matrix $P \in [0,1]^{n \times m}$ such that $\frac{1}{2}(1 - \eta)P\mu \succ \lambda$.

We do not know if $\frac{1}{2}$ is tight

Summary

- We studied conditions guaranteeing general queueing networks’ stability under centralized and decentralized policies, with both impatient and patient queues.
- Conditions for centralized stability in general graphs are natural extensions of those given by Gaitonde & Tardos for complete bipartite graphs
- Conditions for stability under no-regret strategies require new thoughts
  - The dual form of centralized conditions are critical in such extensions
  - For multi-layer graphs, utilities and service priority must be redefined for any PoA type of result; queue lengths are sufficient information.