Ming Hsieh Department of Electrical Engineering

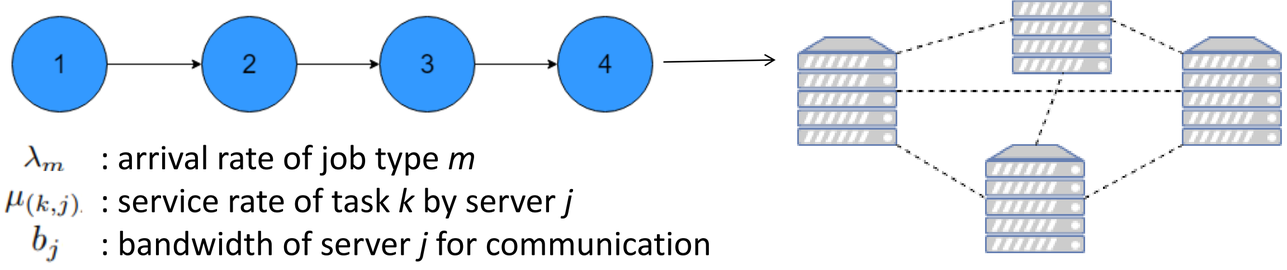
USC Viterbi School of Engineering

# Communication-Aware Scheduling of Serial Tasks for Dispersed Computing

Chien-Sheng Yang[1], Salman Avestimehr[1], Ramtin Pedarsani[2]
[1]University of Southern California [2]University of California, Santa Barbara

## System Model

***M*** different types of jobs where each type is specified by a chain of tasks. Edge of the chain represents a precedence constraint. In the processing network, there are ***J*** servers.




$\lambda_m$ : arrival rate of job type *m*
$\mu_{(k,j)}$ : service rate of task *k* by server *j*
$b_j$ : bandwidth of server *j* for communication
$c_k$ : output size of task *k*

***Definition:*** A scheduling policy is throughput-optimal if it stabilizes the system when there exists a stabilizing scheduling policy.
***Goal:*** Find a throughput optimal scheduling policy.

## Capacity Region

***Capacity region*** is determined by an LP:

$$\text{Maximize} \quad \delta$$

$$\text{subject to} \quad \nu_k \le \sum_{j=1}^{J} \mu_{(k,j)} p_{(k,j)} - \delta, \quad \forall\ 1 \le k \le K$$

$$\frac{b_j q_{(k,j)}}{c_k} \ge \mu_{(k,j)} p_{(k,j)} - \mu_{(k',j)} p_{(k',j)}, \quad \forall\ 1 \le j \le J, \quad \forall (k,k') \in \mathcal{E}$$

$$1 \ge \sum_{k=1}^{K} p_{(k,j)}, \quad \forall\ 1 \le j \le J$$

$$1 \ge \sum_{k \in \{1,2,\ldots,K\} \setminus \mathcal{H}} q_{(k,j)}, \quad \forall\ 1 \le j \le J$$

$$p_{(k,j)} \ge 0,$$
$$q_{(k,j)} \ge 0.$$

Total service capacity on task *k*

Communication constraints

Total capacity of server *j* works on tasks

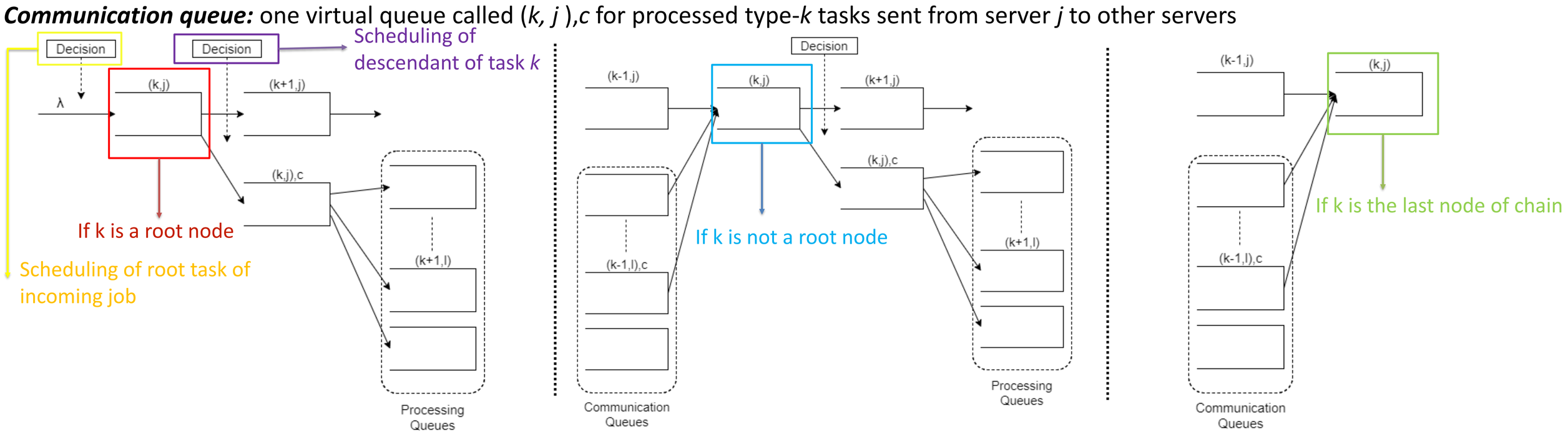
Total capacity of bandwidth server *j* allocates on tasks

Arrival rates are in the capacity region if $\delta^* \ge 0$

## Queuing Network

***Processing queue:*** virtual queue called (*k, j* ) for type-*k* tasks which are processed at server *j*

***Communication queue:*** one virtual queue called (*k, j* ),*c* for processed type-*k* tasks sent from server *j* to other servers




## Equivalent Capacity Region

Corresponding to the queueing network, we define an ***optimization problem***:

$$\text{Maximize} \quad \gamma$$

$$\text{subject to} \quad r_{(k,j)} \le \mu_{(k,j)} p_{(k,j)} - \gamma, \ \forall\ 1 \le j \le J \text{ and } \forall\ 1 \le k \le K.$$

$$r_{(k,j),c} \le \frac{b_j q_{(k,j)}}{c_k} - \gamma, \ \forall\ 1 \le j \le J \text{ and } \forall\ k \in \{1,2,\ldots,K\} \setminus \mathcal{H}.$$

$$1 \ge \sum_{k=1}^{K} p_{(k,j)}, \ \forall\ 1 \le j \le J$$

$$1 \ge \sum_{k=1}^{K} q_{(k,j)}, \ \forall\ 1 \le j \le J$$

$$1 = \sum_{j=1}^{J} u_{m \to j}, \ \forall\ 1 < m < M$$

$$1 = \sum_{l=1}^{J} s_{k,j \to l}, \ \forall\ 1 \le j \le J \text{ and } \forall\ k \in \{1,2,\ldots,K\} \setminus \mathcal{H}.$$

$$p_{kj} \ge 0.$$
$$q_{kj} \ge 0.$$
$$u_{m \to j} \ge 0.$$
$$s_{k,j \to l} \ge 0.$$

Nominal rate to queue (*k, j* )

Nominal rate to queue (*k, j* ),*c*

Fraction of root task of type-*m* job scheduled on server *j*

Fraction that type-*k* tasks are served by server *j* and their descendants are scheduled on server *l*

***Lemma:*** The capacity region characterized by the optimization problem is equivalent to capacity region by the LP.

## Throughput Optimal Policy

***Max-weight policy :***

Given the lengths of virtual queues at time *n*, Max-weight policy allocates the allocation vectors $p$, $q$, $u$ and $s$, i.e.

$$\arg \max_{p,q,u,s \text{ are feasible}} -(Q(n))^T E[\Delta Q(n)|\mathcal{F}^n] - (Q_c(n))^T E[\Delta Q_c(n)|\mathcal{F}^n]$$
$$= \arg \min_{p,q,u,s \text{ are feasible}} (Q(n))^T E[\Delta Q(n)|\mathcal{F}^n] + (Q_c(n))^T E[\Delta Q_c(n)|\mathcal{F}^n],$$

The Max-Weight policy is the choice of $p$, $q$, $u$ and $s$ that minimizes the drift of change of a Lyapunov function:

$$V = \sum_{k,j} Q^2_{(k,j)}(n) + \sum_{k,j} Q^2_{(k,j),c}(n)$$

***Theorem:*** Max-Weight policy for the queueing network is throughput-optimal.

Max-Weight policy is rate stable for all the arrival vectors in the capacity region presented in optimization problem.

Max-weight policy makes the underlying Markov process of the queue-lengths positive recurrent for all the arrival rate vectors that are in the interior of the capacity region.

## Ongoing Work and Future Directions

- Consider jobs modeled as directed acyclic graphs (DAG) and design queueing network with low complexity.
- Simulations on the real world scenario, ex: Amazon EC2, Digital Ocean, etc.

***For more questions, please contact: chienshy@usc.edu***

Ming Hsieh Institute
Ming Hsieh Department of Electrical Engineering