# Scheduling Bags of Non-identical Tasks

Henri Casanova, Matthieu Gallet and Frédéric Vivien

June 2, 2010

# The Problem

- Several bag-of-tasks applications
  (Each application is a collection of similar tasks)
- A master-worker platform
- Objective: maximizing the throughput
- Bad news: a bag is made of similar but not identical tasks

# Presentation outline

Offline Case: Identical Tasks

Offline Case: Tasks With Different Characteristics

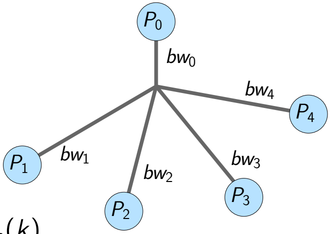
Online Case: Tasks With Different Characteristics

Simulations

# Presentation outline

# Notation

- A master $P_0$ which has an output bandwidth of $bw_0$
- $n$ workers: $P_1$, ..., $P_n$
- Processor $P_i$ has
  - a speed of $s_i$
  - an input bandwidth of $bw_i$
- $m$ bag-of-tasks applications
- Tasks of bag $k$ have
  - a volume of computation of $V_{comp}(k)$
  - a volume of communication of $V_{comm}(k)$
- Communication model: bounded multi-port with linear communication times



# Constraints

1. Cumulative throughput of $T_k$:

$$\rho^{(k)} = \sum_{1 \leq i < n} \rho_i^{(k)}$$

2. Throughput of $T_k$ proportional to its priority:

$$\frac{\rho^{(k)}}{\pi_k} = \frac{\rho^{(1)}}{\pi_1}$$

Objective

$$\textsc{Maximize} \quad \rho^{(1)}$$

# Constraints (continued)

3. Constraint on computation capabilities of worker $P_i$

$$\sum_{1 \le k \le m} \rho_i^{(k)} \frac{V_{comp}(k)}{s_i} \le 1$$

4. Constraint on communication capabilities of worker $P_i$

$$\sum_{1 \le k \le m} \rho_i^{(k)} \frac{V_{comm}(k)}{bw_i} \le 1$$

5. Constraint on communication capabilities of the master

$$\sum_{1 \le i < n} \sum_{1 \le k \le m} \rho_i^{(k)} \frac{V_{comm}(k)}{bw_0} \le 1$$

# Complete Linear Program

$$\left\{\begin{array}{l}
\textsc{Maximize } \rho^{(1)} \textsc{ under the constraints} \\
\quad \forall k \in [1, m], \quad \displaystyle\sum_{1 \le i < n} \rho_i^{(k)} = \rho^{(k)} \\
\quad \forall k \in [1, m], \quad \dfrac{\rho^{(k)}}{\pi_k} = \dfrac{\rho^{(1)}}{\pi_1} \\
\quad \forall i \in [1, n], \quad \displaystyle\sum_{1 \le k \le m} \rho_i^{(k)} \frac{V_{comp}(k)}{s_i} \le 1 \\
\quad \forall i \in [1, n], \quad \displaystyle\sum_{1 \le k \le m} \rho_i^{(k)} \frac{V_{comm}(k)}{bw_i} \le 1 \\
\qquad \displaystyle\sum_{1 \le i < n} \sum_{1 \le k \le m} \rho_i^{(k)} \frac{V_{comm}(k)}{bw_0} \le 1
\end{array}\right.$$

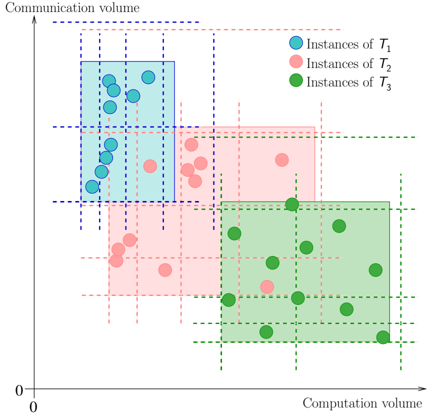
# Presentation outline

# Notation

- A master $P_0$ which has an output bandwidth of $bw_0$
- $n$ workers: $P_1$, ..., $P_n$
- Processor $P_i$ has
  - a speed of $s_i$
  - an input bandwidth of $bw_i$
- $m$ bag-of-tasks applications
- Tasks of bag $k$ have
  - $X_{comm}^{(k)}$ is a random variable
    the $u$-th instance has a communication volume of $X_{comm}^{(k)}(u)$
    $\min_{comm}^{(k)} \leq X_{comm}^{(k)}(u) \leq \max_{comm}^{(k)}$
  - $X_{comp}^{(k)}$ is a random variable
    the $u$-th instance has a computation volume of $X_{comp}^{(k)}(u)$
    $\min_{comp}^{(k)} \leq X_{comp}^{(k)}(u) \leq \max_{comp}^{(k)}$
- Communication model:
  bounded multi-port with linear communications times

# An $\varepsilon$-approximation scheme

Underlying principle: split each application into several virtual applications in which two instances only have small differences in term of communication and computation volumes.



# Formal splitting

$$\gamma_q^{(k)} = (1+\varepsilon)^q \min_{comp}^{(k)}, \text{ with } 0 \leq q \leq Q^{(k)} = 1 + \left\lfloor \frac{\ln\left(\frac{\max_{comp}^{(k)}}{\min_{comp}^{(k)}}\right)}{\ln(1+\varepsilon)} \right\rfloor$$

$$\delta_r^{(k)} = (1+\varepsilon)^r \min_{comm}^{(k)}, \text{ with } 0 \leq r \leq R^{(k)} = 1 + \left\lfloor \frac{\ln\left(\frac{\max_{comm}^{(k)}}{\min_{comm}^{(k)}}\right)}{\ln(1+\varepsilon)} \right\rfloor$$

Instance $u$ of $T_k$ belongs to $I_{q,r}^{(k)} = \left[\gamma_q^{(k)}; \gamma_{q+1}^{(k)}\right] \times \left[\delta_r^{(k)}; \delta_{r+1}^{(k)}\right]$ if

- $\gamma_q^{(k)} \leq X_{comp}^{(k)}(u) \leq \gamma_{q+1}^{(k)}$ and
- $\delta_r^{(k)} \leq X_{comm}^{(k)}(u) \leq \delta_{r+1}^{(k)}$

# Virtual applications

- Instances of $T_k$ in $I_{q,r}^{(k)}$ define virtual application $T_{k,q,r}$
- $p_{q,r}^{(k)}$ probability of an instance of $T_k$ to belong to virtual application $T_{k,q,r}$:

$$p_{q,r}^{(k)} = \mathcal{P}\left(\gamma_q^{(k)} \leq X_{comp}^{(k)} < \gamma_{q+1}^{(k)}; \delta_r^{(k)} \leq X_{comm}^{(k)} < \delta_{r+1}^{(k)}\right)$$

$$\forall k, \sum_{q,r} p_{q,r}^{(k)} = 1$$

- $\rho_{i,q,r}^{(k)}$: contribution of processor $P_i$ to the throughput of virtual application $T_{k,q,r}$
- Throughput of virtual application $T_{k,q,r}$ is related to the throughput of $T_k$:

$$\forall k, \forall q < Q^{(k)}, \forall r < R^{(k)}, \sum_{1 \leq i < n} \rho_{i,q,r}^{(k)} = p_{q,r}^{(k)} \rho^{(k)}$$

# Transposing the constraints

- Throughput of $T_k$ is still proportional to its priority:

$$\forall k \in [1, m],\ \frac{\rho^{(k)}}{\pi_k} = \frac{\rho^{(1)}}{\pi_1}$$

- Constraint on computation capabilities of worker $P_i$
  Problem: We do not know the execution time of instances
  Solution: We (conservatively) over-approximate them

$$\forall i \in [1, n],\ \sum_{k=1}^{m} \sum_{\substack{q<Q^{(k)} \\ r<R^{(k)}}} \left( \rho_{i,q,r}^{(k)} \frac{\gamma_{r+1}^{(k)}}{s_i} \right) \leq 1$$

# Transposing the constraints (cont.)

- Constraint on communication capabilities of worker $P_i$

$$\forall 1 \leq i < n, \sum_{k=1}^{m} \sum_{\substack{q<Q^{(k)} \\ r<R^{(k)}}} \left( \rho_{i,q,r}^{(k)} \frac{\delta_{r+1}^{(k)}}{bw_i} \right) \leq 1$$

- Constraint on communication capabilities of the master

$$\sum_{k=1}^{m} \sum_{\substack{q<Q^{(k)} \\ r<R^{(k)}}} \left( \rho_{i,q,r}^{(k)} \frac{\delta_{r+1}^{(k)}}{bw_0} \right) \leq 1$$

# New linear program

$$
\left\{
\begin{array}{l}
\textsc{Maximize } \rho = \rho^{(1)} \textsc{ under the constraints} \\
\quad \forall k \in [1,m], \forall q < Q^{(k)}, \forall r < R^{(k)}, \quad \sum_{i=1}^{n} \rho_{i,q,r}^{(k)} = p_{q,r}^{(k)} \rho^{(k)} \\
\quad \forall k \in [1,m], \quad \dfrac{\rho^{(k)}}{\pi_k} = \dfrac{\rho^{(1)}}{\pi_1} \\
\quad \forall i \in [1,n], \quad \sum_{k=1}^{m} \sum_{\substack{q<Q^{(k)} \\ r<R^{(k)}}} \left( \rho_{i,q,r}^{(k)} \dfrac{\gamma_{q+1}^{(k)}}{s_i} \right) \leq 1 \\
\quad \forall i \in [1,n], \quad \sum_{k=1}^{m} \sum_{\substack{q<Q^{(k)} \\ r<R^{(k)}}} \left( \rho_{i,q,r}^{(k)} \dfrac{\delta_{r+1}^{(k)}}{bw_i} \right) \leq 1 \\
\qquad \sum_{i=1}^{n} \sum_{k=1}^{m} \sum_{\substack{q<Q^{(k)} \\ r<R^{(k)}}} \left( \rho_{i,q,r}^{(k)} \dfrac{\delta_{r+1}^{(k)}}{bw_0} \right) \leq 1
\end{array}
\right.
$$

# Performance

**Theorem.**

An optimal solution of the Linear Program describes a solution with a throughput $\rho$ larger than $\rho^*/(1+\varepsilon)$ (with a great probability), where $\rho^*$ is the optimal throughput.

# Presentation outline

# Aim

- Non-clairvoyant about computation volumes
- Communication volumes can be supposed to be known
- Underlying distributions are unknown

Is there any hope?

# Case with dominant computations

**Theorem.**

On-Demand policy is asymptotically optimal when

- Computations are always dominant:

$$\forall i \in [1, n], \qquad \min_{k,u} \frac{X^{(k)}_{comp}(u)}{s_i} \quad \geq \quad \max_{k',u'} \frac{X^{(k')}_{comm}(u')}{bw_i}$$

- The master's bandwidth is not constraining:

$$bw_0 \geq \sum_{i=1}^{n} bw_i$$

- Each worker as a limited number of buffers ($\in [2, n_{buffers}]$)

# Case with infinite buffers

**Theorem.**

On-Demand has no constant competitive ratio

- 1 application with $N$ tasks and unitary communication and computation volume, master's bandwidth not constraining
- $bw_1 = \frac{1}{2N}; bw_2 = ... = bw_n = 1$
- $s_1 = 2(n-1)N; s_2 = ... = s_n = 1$
- Possible schedule: ignore worker $P_1$:
  $makespan_{opt} \leq \left\lceil \frac{N}{n-1} \right\rceil + 1$
- solution of On-Demand 1 task each for $P_2$, ..., $P_n$,
  $N-(n-1)$ tasks for $P_1$.
  $Makespan_{\text{On-Demand}} \geq (N-(n-1))s_1 \geq N \times Makespan_{opt}$
  (for $N \geq 4n$).

# Case with dominant communications

**Theorem.**

On-Demand policy is asymptotically optimal when

- Communications are always dominant:

$$\forall i \in [1, n], \qquad \max_{k,u} \frac{X^{(k)}_{comp}(u)}{s_i} \quad \leq \quad \min_{k',u'} \frac{X^{(k')}_{comm}(u')}{bw_i}$$

- Each worker has a limited number of buffers ($\in [2, n_{buffers}]$)

# Practical heuristics

- Use the first 10% of instances to gather data on applications
- From this sample, split applications into virtual applications
  - arithmetical buckets
  - geometrical buckets
  - recursive buckets

  (We only report on Geometrical buckets has they lead to (slightly) better results)
- Apply the multi-application linear program on the virtual applications (with the rounding used for tasks with different characteristics)
- Schedule realized using a 1D load-balancing among processors (per virtual application)

# Presentation outline

# Simulation settings

- 3 or 4 applications
- 100, 1000, or 5000 instances per application
- Communication volume uniformly picked in $[\min_{comm}; \max_{comm}]$ with $\max_{comm} / \min_{comm}$ in $\{1, 1.35, 1.65, 2.35, 2.65\}$.
- *Correlation factor* $\phi \in [0, 1]$ (0: no correlation).
  For instance $u$: $\exists \lambda, X_{comm}^{(k)}(u) = \lambda \min_{comm}^{(k)} + (1 - \lambda) \max_{comm}^{(k)}$
  $V_{comp}(i)$ is randomly picked in

$$\Big[(\phi\lambda + 1 - \phi) \min_{comp}^{(k)} + \phi(1 - \lambda) \max_{comp}^{(k)}, \\ \phi\lambda \min_{comp}^{(k)} + (1 - \lambda\phi) \max_{comp}^{(k)}\Big]$$

- Platforms: 3, 5, 10, or 15 workers.
  Master's bandwidth = 1, 5, or 100 times the average bandwidth of workers

# Overall results

| Heuristic | Normalized to best | Normalized to UB |
|---|---|---|
| On-Demand | 0.87 ($\sigma = 0.108$) | 0.821 ($\sigma = 0.109$) |
| Round-Robin | 0.779 ($\sigma = 0.123$) | 0.736 ($\sigma = 0.126$) |
| LP_samp(ARITH, 1, 1) | 0.971 ($\sigma = 0.0362$) | 0.917 ($\sigma = 0.0651$) |
| LP_samp(GEOM, 2, 1) | 0.875 ($\sigma = 0.106$) | 0.829 ($\sigma = 0.122$) |
| LP_samp(GEOM, 4, 1) | 0.819 ($\sigma = 0.13$) | 0.777 ($\sigma = 0.144$) |
| LP_samp(GEOM, 8, 1) | 0.795 ($\sigma = 0.136$) | 0.754 ($\sigma = 0.149$) |
| LP_samp(GEOM, 2, 2) | 0.842 ($\sigma = 0.129$) | 0.799 ($\sigma = 0.144$) |
| LP_samp(GEOM, 4, 4) | 0.812 ($\sigma = 0.139$) | 0.771 ($\sigma = 0.153$) |
| 0.05-approx | 0.993 ($\sigma = 0.022$) | 0.937 ($\sigma = 0.0555$) |
| 0.2-approx | 0.985 ($\sigma = 0.0201$) | 0.93 ($\sigma = 0.0513$) |

# Conclusion

- Always worth to distinguish applications
- Further splitting worthwhile if
  - Lots of instances
  - Comparable communication and computation costs
  - Communication-to-computation ratio depends of communication volume