Real-time GDA

ABSTRACT

ACM Reference format:
. 2016. Real-time GDA. In *Proceedings of ACM Conference, Washington, DC, USA, July 2017 (Conference'17),* 3 pages.
DOI: 10.475/123_4

1 SINGLE DAG QUERY

1.1 Model

1.1.1 Wide Area Network. Let $\mathcal{S}$ be the set of sites that hold data and run tasks, and $\mathcal{L}$ be the set of directed edges that represent inter-site links. For each inter-site WAN link $(i, j) \in \mathcal{L}$, let B_{ij} be the bandwidth and C_{ij} be the cost to transfer one unit of data from site i to site j.

Assumption 1. *The bandwidths are stable within the time frame of real-time data analytics.*

For the analysis we make the following assumption:

Assumption 2. *Data transfers on a particular link are non-overlapping.* *(To do: Show that this assumption does not add suboptimality to our objective, i.e., any overlapping schedule can be transformed into a non-overlapping schedule without any loss to the optimal value.)*

Assumption 3. *Data transfers on a particular link are non-interruptible.* *(To do: Show that this assumption does not add suboptimality to our objective, i.e., any schedule with interruptions can be transformed into a non-interruptible schedule without any loss to the optimal value.)*

1.1.2 DAG of tasks. We define a stage in the DAG as a group of tasks that have the same input data dependencies. Let $\mathcal{T}$ be the set of stages and $\mathcal{D}$ be the directed set of edges that represent stage dependencies for the DAG, respectively. Some of the stages do not have any incoming edges and so represent the locations of the raw input data. Let $\mathcal{R} \subset \mathcal{T}$ be the set of stages for the raw input data. The final stage in the DAG does not have any outgoing edges and so represent the final destination of the query's response denoted by $F \in \mathcal{T}$. Each stage dependency $(k, l) \in \mathcal{D}$ also has a corresponding amount of data D^{kl} that must be transfered from stage k to stage l.

Assumption 4. *A stage must have completely received all of its input data before it can process and start transferring data to another stage.*

The DAG also has a start time T_0 and a finish time T_f for which the schedule of data transfers must respect.

1.1.3 Stage assignment decisions. We model each stage as a group of tasks that must be scheduled to run together at the same site. The decision of stage k to be placed at site i is represented by the binary decision variable $x_i^k \in \{0, 1\}$. This means that for any directed edge $(k, l) \in \mathcal{D}$, then $D^{kl} x_i^k x_j^l$ of data will be transfered across link $(i, j) \in \mathcal{L}$. Note that the set of stages $\mathcal{R}$ and F which respectively represent the raw data and the DAG's final stage have decision variables which are preset according to its site-wise distribution and final stage location.

Each stage dependency (k, l) and pair of links $(i, j) \in \mathcal{L}$ have a binary decision variable $q_{ij}^{kl} \in \{0, 1\}$ that determines whether link (i, j) is available to be used by stage (k, l) or not.

1.1.4 Scheduling decisions. For each link, an ordering needs to be decided between all the data transfers on that link. Let $u_{ij}^{kl|mn} \in \{0, 1\}$ be the binary decision to schedule data transfers (k, l) before (m, n) on link (i, j). Note that this decision variable is already set if there exists a directed path that can pass through both (k, l) and (m, n).

1.2 Stage Assignment Problem

1.2.1 Problem Statement. Given the available links for each dependency we have the following problem to minimize the cost of the WAN by deciding where to place each stage:

$$\min_{\mathbf{x}} \quad \sum_{(k,l)\in\mathcal{D}} \sum_{(i,j)\in\mathcal{L}} C_{ij} D^{kl} x_i^k x_j^l$$

$$\text{s.t.} \quad x_i^k x_j^l \le q_{ij}^{kl} \quad \forall (i,j) \in \mathcal{L}, \forall (k,l) \in \mathcal{D} \tag{1a}$$

$$\sum_{i\in\mathcal{S}} x_i^k = 1 \quad \forall k \in \mathcal{T} \tag{1b}$$

$$x_i^k \in \{0,1\} \quad \forall i \in \mathcal{S}, k \in \mathcal{T} \tag{1c}$$

Note that for any stage $k \in \mathcal{R} \cup F$, the placement decision is preset.

1.3 Task Assignment Problem

1.3.1 Problem Statement. We model each stage as a distributable group of tasks that may be fractionally assigned to multiple sites. The fraction of tasks from stage k placed at site i is represented by the nonnegative decision variable x_i^k and proportionally determines the size of data transfers. This means that for any directed edge $(k, l) \in \mathcal{D}$, then $D^{kl} x_i^k x_j^l$ of data will be transfered across link $(i, j) \in \mathcal{L}$. Note that the set of stages $\mathcal{R}$ which represent the raw data have a decision variable which is preset according to its site-wise distribution. Each data transfer is given an upper bound on its size d_{ij}^{kl} which will be used by the scheduling algorithm.

The goal of the task assignment problem is to fractionally distribute the stages of tasks to minimize the WAN usage for a given set of link duration upper bounds:

$$\min_{\mathbf{x}} \quad \sum_{(k,l)\in\mathcal{D}} \sum_{i\in\mathcal{S}} \sum_{j\in\mathcal{S}} C_{ij} D^{kl} x_i^k x_j^l$$

$$\text{s.t.} \quad D^{kl} x_i^k x_j^l \le d_{ij}^{kl} \quad \forall \{i,j\} \in \mathcal{S}, \forall (k,l) \in \mathcal{D} \tag{2a}$$

$$\sum_{i\in\mathcal{S}} x_i^k = 1 \quad \forall k \in \mathcal{T} \tag{2b}$$

$$x_i^k \ge 0 \quad \forall i \in \mathcal{S}, k \in \mathcal{T} \tag{2c}$$

1.3.2 Necessary Feasibility Conditions. There needs to be a minimal amount of data flow allowed. Summing the over all pairs of

Conference'17, Washington, DC, USA

DOI: 10.475/123_4

sites for each $(k,l) \in \mathcal{D}$:

$$\sum_{i\in\mathcal{S}}\sum_{j\in\mathcal{S}} D^{kl}x_i^k x_j^l \le \sum_{i\in\mathcal{S}}\sum_{j\in\mathcal{S}} d_{ij}^{kl} \quad \forall (k,l)\in\mathcal{D}.$$

Since the sum of the x's are 1 we have:

$$D^{kl} \le \sum_{i\in\mathcal{S}}\sum_{j\in\mathcal{S}} d_{ij}^{kl} \quad \forall (k,l)\in\mathcal{D}. \tag{3}$$

1.3.3 *Sufficient Feasibility Conditions.* (To do:)

1.3.4 *Convexifying the problem.*

We are interested in convexifying the function $x_i^k x_j^l$.

When there are not linear equalities, then paper [2] gives necessary and sufficient properties of functions that can be used to replace the product $x_i^k x_j^l$. The idea is to replace x_i^k with $f_1(y_1)$ and x_j^l with $f_2(y_2)$ so that $F(y_1,y_2) := f_1(y_1)f_2(y_2)$ is jointly convex in y_1 and y_2. The properties they give imply that both $f_1(y_1)$ and $f_2(y_2)$ need to be strictly convex which works well for the objective function of (2) and (2a) but not for (2b) when it becomes the summation of strictly convex functions.

However when there are linear equalities that we want to keep, [3] uses $x_i^k := f_i^k(y_i^k) = e^{y_i^k} - \tau$ and then turns $x_i^k x_j^l$ into $e^{y_i^k+y_j^l} - \tau(y_i^k + y_j^l) + \tau^2$. The caveat is that we now have the equalities $y_i^k = \ln(x_i^k + \tau)$ which are concave and not linear. The paper uses a piecewise linear function to under-estimate the equalities. They run it on a convex optimization solver and update the piecewise linear grid based on the previous iteration's solution. This will numerically converge to the global optimal.

1.3.5 *A convex lower bound on $x_i^k x_j^l$.*

and is tight if the decision variables x_i^k are binary:

$$x_i^k x_j^l \ge \max\{0, x_i^k + x_j^l - 1\} \quad \forall x_i^k \in [0,1], x_j^l \in [0,1] \tag{4}$$

$$x_i^k x_j^l = \max\{0, x_i^k + x_j^l - 1\} \quad \forall x_i^k \in \{0,1\}, x_j^l \in \{0,1\} \tag{5}$$

1.3.6 *A convex upper bound on $x_i^k x_j^l$.*

$$x_i^k x_j^l \le \max\{(x_i^k)^2, (x_j^l)^2\} = (\max\{x_i^k, x_j^l\})^2 \quad \forall x_i^k \in [0,1], x_j^l \in [0,1] \tag{6}$$

1.3.7 *Characterizing the optima* (2) *directly.*

Although Problem (2) is not convex, we can still find properties of the optimal solution.

By taking the Lagrange dual and the dual variables $(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\phi})$ corresponding with the respective constraints, we have the following first-order necessary stationary conditions for optimality:

$$\sum_{k:(k,l)\in\mathcal{D}}\sum_{i\in\mathcal{S}} C_{ij}D^{kl}x_i^k + \sum_{k:(l,k)\in\mathcal{D}}\sum_{i\in\mathcal{S}} C_{ji}D^{lk}x_i^k + \sum_{k:(k,l)\in\mathcal{D}}\sum_{i\in\mathcal{S}} \lambda_{ij}^{kl}D^{kl}x_i^k + \sum_{k:(l,k)\in\mathcal{D}}\sum_{i\in\mathcal{S}} \lambda_{ji}^{lk}D^{lk}x_i^k - \mu^l - \phi_j^l = 0 \quad \forall j\in\mathcal{S}, \forall l\in\mathcal{T}$$

Combine the 1st and 3rd summations, combine the 2nd and 4th summations:

$$\sum_{k:(k,l)\in\mathcal{D}}\sum_{i\in\mathcal{S}} D^{kl}(C_{ij}+\lambda_{ij}^{kl})x_i^k + \sum_{k:(l,k)\in\mathcal{D}}\sum_{i\in\mathcal{S}} D^{lk}(C_{ji}+\lambda_{ji}^{lk})x_i^k - \mu^l - \phi_j^l = 0 \quad \forall j\in\mathcal{S}, \forall l\in\mathcal{T}$$

Relabel the index in the 2nd summation of the second summation set, and flip the summation order:

$$\sum_{i\in\mathcal{S}}\sum_{k:(k,l)\in\mathcal{D}} D^{kl}(C_{ij}+\lambda_{ij}^{kl})x_i^k + \sum_{i\in\mathcal{S}}\sum_{m:(l,m)\in\mathcal{D}} D^{lm}(C_{ji}+\lambda_{ji}^{lm})x_i^m = \mu^l + \phi_j^l \quad \forall j\in\mathcal{S}, \forall l\in\mathcal{T} \tag{7}$$

The dual feasibility constraints give us:

$$\lambda_{ij}^{kl} \ge 0 \quad \forall \{i,j\}\in\mathcal{S}, \forall (k,l)\in\mathcal{D} \tag{8a}$$

$$\phi_j^l \ge 0 \quad \forall j\in\mathcal{S}, \forall l\in\mathcal{T} \tag{8b}$$

The complementary slackness constraints for the inequalities give us:

$$\lambda_{ij}^{kl}\left(D^{kl}x_i^k x_j^l - d_{ij}^{kl}\right) = 0 \quad \forall \{i,j\}\in\mathcal{S}, \forall (k,l)\in\mathcal{D} \tag{9a}$$

$$\phi_j^l x_j^l = 0 \quad \forall j\in\mathcal{S}, \forall l\in\mathcal{T} \tag{9b}$$

Therefore the necessary conditions for optimality are (2a) (2b) (2c) (7) (8) (9).

Inferences from the conditions

(1) At the optimal solution, the dual variables $\boldsymbol{\lambda}$ correspond with the negative gradient of the objective function w.r.t. the data transfer upper limits $\mathbf{d}$. (See [1] Proposition 3.3.3)
(2) At the optimal solution, μ^l is the total WAN cost plus the product of dual prices and the upper limits for all incoming and outgoing data transfers associated with stage l. Also each fraction of stage l at site j must share that same fraction of that stage's total cost. Multiply (7) by x_j^l and substitute with (9):

$$\sum_{i\in\mathcal{S}}\sum_{k:(k,l)\in\mathcal{D}} C_{ij}D^{kl}x_i^k x_j^l + \lambda_{ij}^{kl}d_{ij}^{kl} + \sum_{i\in\mathcal{S}}\sum_{m:(l,m)\in\mathcal{D}} C_{ji}D^{lm}x_j^l x_i^m + \lambda_{ji}^{lm}d_{ji}^{lm} = \mu^l x_j^l \quad \forall j\in\mathcal{S}, \forall l\in\mathcal{T} \tag{10}$$

Then sum for all $j \in \mathcal{S}$ and then apply (2b) to the RHS:

$$\sum_{j\in\mathcal{S}}\sum_{i\in\mathcal{S}}\sum_{k:(k,l)\in\mathcal{D}} C_{ij}D^{kl}x_i^k x_j^l + \lambda_{ij}^{kl}d_{ij}^{kl} + \sum_{j\in\mathcal{S}}\sum_{i\in\mathcal{S}}\sum_{m:(l,m)\in\mathcal{D}} C_{ji}D^{lm}x_j^l x_i^m + \lambda_{ji}^{lm}d_{ji}^{lm} = \mu^l \quad \forall l\in\mathcal{T} \tag{11}$$

1.3.8 *Optimality conditions if predecessor and successor stages are pre-assigned.* (To do:)

1.3.9 *Take advantage of sparcity.* (To do:)

1.4 Data Transfer Scheduling Problem

Because of the dependencies specified by the DAG, each data transfer upper bound d_{ij}^{kl} must be scheduled after the deadline of stage k denoted t^k and before the deadline of stage l denoted t^l. Therefore the goal is to find a scheduling policy π and stage deadlines $\mathbf{t}$ such that the query start time T_0 and deadline T_f are satisfied for the given set of durations $\mathbf{d}$ and stage dependencies $\mathcal{D}$.

1.5 Jointly optimization

Both the Task Assignment and Data Transfer Scheduling problems are connected through the data transfer upper bounds **d**. Therefore the joint optimization problem becomes if we can find a feasible schedule that minimizes the WAN usage cost by adjusting **d**.

REFERENCES

[1] Dimitri P Bertsekas. 1999. *Nonlinear programming*. Athena scientific Belmont.
[2] CE Gounaris and CA Floudas. 2008. Convexity of products of univariate functions and convexification transformations for geometric programming. *Journal of Optimization Theory and Applications* 138, 3 (2008), 407–427.
[3] Ray PöRn, Kaj-Mikael BjöRk, and Tapio Westerlund. 2008. Global solution of optimization problems with signomial parts. *Discrete optimization* 5, 1 (2008), 108–120.

APPENDIX