A Network Flow Approach to Bike Sharing

Fatih Bakir, Gabriel Siqueira, Goksu Guvendiren, Mert Kosan, Richika Sharan, Zexi Huang*†
Department of Computer Science
University of California, Santa Barbara

December 7, 2018

1 Notations

1.1 Model parameters (given)

1. n: number of locations.
2. k: number of time slots in a day.
3. K: number of days considered in our model.
4. b: cost to move a bike to another location by us.
5. m: number of bikes.

1.2 Input variables (also given)

1. $s_{i,t}$: supply at ith location at tth time slot, i.e., number of bikes arriving at ith location at tth time slot due to riders. $s_{i,1}$ is the initial number of bikes at each location. Note that $s_{i,t} \geq 0, \forall i, t$.
2. $d_{i,t}$: demand at ith location at tth time slot, i.e., number of bikes leaving at ith location at tth time slot due to riders.
3. Implicit constraints of demands and supplies: since the total number of bikes should conserve at different time slots, the total number of bikes leaving at tth time slot should be equal to the total number of bikes arriving at $(t+1)$th time slot, which is

$$\sum_{i=1}^{n} d_{i,t} = \sum_{i=1}^{n} s_{i,t+1}, t \geq 1 \tag{1.1}$$

*All group members contribute equally to this project.
†Correspondence should be addressed to Z. Huang. E-mail: zexi_huang@cs.ucsb.edu.

4. **Remarks**: compared to LP formulation, the supply in LP $s_{i,t}^{LP}$ is actually the difference between supply at t $s_{i,t}$ and the demand at $t-1$ $d_{i,t-1}$, that is

$$s_{i,t}^{LP} = s_{i,t} - d_{i,t-1} \tag{1.2}$$

We can see that $s_{i,t}^{LP}$ is the net value of bikes arriving at ith location at tth time slot. Therefore, $s_{i,t}^{LP}$ can be either positive and negative.

1.3 Decision variables (to be optimized)

1. $x_{i,j,t}$: number of bikes changed by us, from ith location to jth location, at tth time slot. Note that $x_{i,j,t} \neq 0$ only when $t = Nk/2 + 1, N = 0, 1, 2, ..., 2K - 1$.

1.4 Intermediate variables (based on dynamics)

1. $l_{i,t}$: number of leftover bikes at ith location at tth time slot. The dynamics of this quantity is

$$l_{i,t} = l_{i,t-1} + s_{i,t} - d_{i,t} + \sum_{w=1}^{n} x_{w,i,t} - \sum_{w=1}^{n} x_{i,w,t}, t \geq 2 \tag{1.3}$$

where $l_{i,t-1}$ is the number of leftover bikes at previous time slot, $s_{i,t}$ is number of bikes arriving due to riders, $d_{i,t}$ is the number of bikes leaving, and $\sum_{w=1}^{n} x_{w,i,t}$ (or $\sum_{w=1}^{n} x_{i,w,t}$) is the number of bikes moving in (or out) ith location by us at tth time slot. Note that the initial number of leftover bikes $l_{i,0}$ are set to zero.

2. **Remarks**: compared to the dynamics of number of bikes $a_{i,t}^{LP}$ in the LP formulation, $l_{i,t}$ is actually the number of bikes $a_{i,t}^{LP}$ minus the number of demanded bikes $d_{i,t}$, that is,

$$l_{i,t} = a_{i,t}^{LP} - d_{i,t} \tag{1.4}$$

By recursive substitution of $l_{i,t}$ in Equation 1.3, we can derive similar dynamics of $a_{i,t}^{LP}$, as Equation 2.1 of [**CS 231 Project: LP Formulation**] .

1.5 Objective function

We want to minimize the total cost of our relocation.

$$\min \sum_{i}^{n} \sum_{j}^{n} \sum_{t=1}^{Kk} b x_{i,j,t} \tag{1.5}$$

2 Network flow model

We want to formulate the problem with a network flow model. Basically, the nodes should represent different locations at different time slots, and the edges should capture the movement of bikes across different locations and time slots.

2.1 Naive model

A naive way to formulate the problem is as follows and shown in Figure 1.

1. Nodes:
   (a) Location nodes: Use one node $v_{i,t}$ to represent ith location at tth time slot.
   (b) Super nodes: one super source node s and one super sink node t.
2. Edges:
   (a) Edges with super nodes: An edge connects from s to every $v_{i,1}$, with capacity $s_{i,1}$ to capture the initial number of bikes at each location. An edge connects from $v_{i,Kk}$ to t, with infinite capacity, to capture the final number of bikes at each location.
   (b) Inter-layer movement edges: Bikes at ith location at tth time slot only has two possible movements due to the riders: either as leftover bikes at ith location at $t+1$th time slot, or as the leaving bikes to reach other locations j at $t+1$th time slot. We capture this dynamics with an edge from every $v_{i,t}$ to every $v_{i,t+1}$, with infinite capacity.
   (c) Intra-layer movement edges: Bikes at ith location at tth time can also be changed by us. We connect every pairs of nodes at possible time slots $t = Nk/2+1, N = 0, 1, 2, ..., 2K-1$, with an infinite capacity. Now note that these edges also come with a unit price b, as our cost of movement. All other edges mentioned before have a price of 0.
3. Flow: flow on each edge represent the number of bikes moved between different time slots with different means:
   (a) Flow on edges with super nodes capture the initial number of bikes at each location and the final number of bikes at each location.
   (b) Flow on inter-layer movement edges capture the number of bikes moved by riders.
   (c) Flow on intra-layer movement edges capture the number of bikes moved by us.

Then, a flow from s to t is a possible movement of bikes across different locations and different time slots, as it capture the dynamics as specified by Equation 1.3, with flow conservation property.

However, in order to faithfully model the problem, there are some additional constraints that can't be captured with the capacity constraints in the network. More specifically, we have

1. Supplying constraints: number of bikes arriving at $v_{i,t+1}$ from $v_{j,t}, j \neq i$ (i.e., flow on the respective edges), should sum up to the supply at $v_{i,t+1}$, i.e., $s_{i,t+1}$.
2. Demanding constraints: number of bikes leaving from $v_{i,t}$ to $v_{j,t+1}, j \neq i$ (i.e., flow on the respective edges), should sum up to the demand at $v_{i,t}$, i.e., $d_{i,t}$.

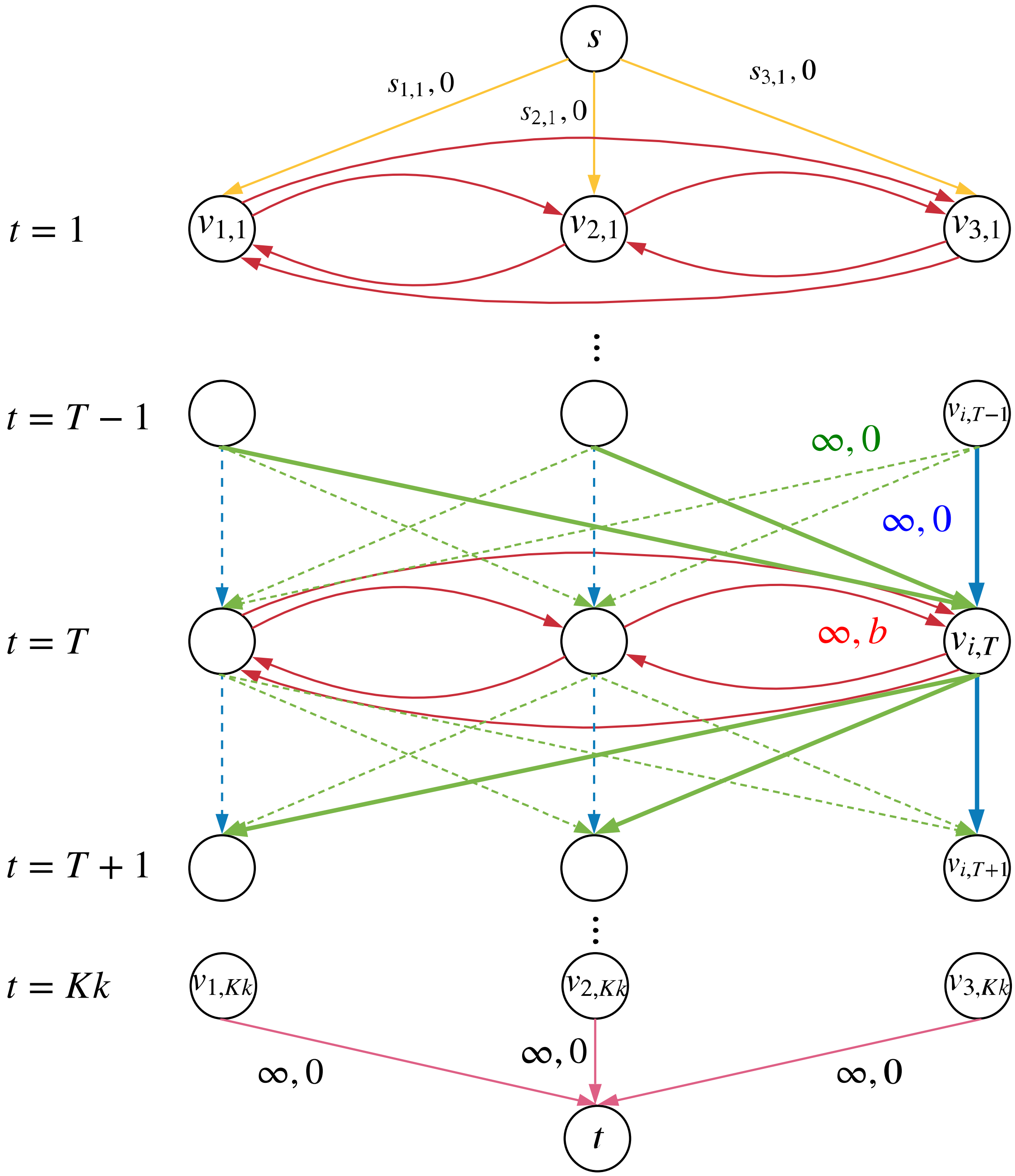

Figure 1: The diagram for the basic model.

Now, if a $s-t$ flow on the network with value equal to the sum of initial number of bikes $m = \sum_{i=1}^{n} s_{i,1}$, can also satisfy these additional constraints, we find a feasible solution of the movement of bikes.

Alternatively, we can also formulate the network flow as a mincost flow problem, by treating s as the only supply node with $d_s = -\sum_{i=1}^{n} s_{i,1}$, t as the only demand node with $d_t = \sum_{i=1}^{n} s_{i,1}$, and all other nodes as transshipment nodes with $d_v = 0$. Then, a mincost flow on this network satisfying the additional constraints is the optimal solution of the bike sharing problem.

2.1.1 Proof of correctness

We show the correctness of our model by showing:

1. Our model captures all the constraints of the original problem.

    (a) The supplying and demanding constraints are captured by external flow constraints on the corresponding set of edges:

    $$s_{i,t} = \sum_{j \neq i} f_{v_{j,t-1}, v_{i,t}} \tag{2.1}$$

    $$d_{i,t} = \sum_{j \neq i} f_{v_{i,t}, v_{j,t+1}} \tag{2.2}$$

    (b) The dynamics of leftover bikes Equation 1.3 is equivalent to the flow conservation property of at each node:

    $$s_{i,t} + l_{i,t-1} + \sum_{j \neq i} f_{v_{i,t}, v_{j,t}} = d_{i,t} + l_{i,t} + \sum_{j \neq i} f_{v_{j,t}, v_{i,t}} \tag{2.3}$$

2. The objective of our model, i.e., to find a mincost of the flow, is exactly the same objective of the original problem, as in Equation 1.5. This is based on the observation that since only intra-layer edges have non-zero price, to minimize the cost of flow on the network is to minimize the cost of flow on intra-layer edges, which is

    $$L = \sum_{i}^{n} \sum_{j}^{n} \sum_{t=1}^{Kk} b f_{v_{i,t}, v_{j,t}} \tag{2.4}$$

    This objective is the same as Equation 1.5, when we denote $f_{v_{i,t}, v_{j,t}}$ as $x_{i,j,t}$.

2.2 Three-node model

In order to incorporate the constraints in the model, we consider the following tricks:

1. For each node $v_{i,t}$, replace it with three nodes $v_{i,t}^{in}$, $v_{i,t}^{left}$ and $v_{i,t}^{out}$. All these nodes come with zero demands.

2. Edges with super node s are connected to $v_{i,1}^{in}$ with same price and capacity. Edges with super node t are connected to $v_{i,Kk}^{left}$ and $v_{i,Kk}^{out}$, each with infinite capacity and zero price.

3. Inter-layer edges capturing arriving bikes from $v_{i,t}$ to $v_{j,t+1}, j \neq i$ are connected from $v_{i,t}^{out}$ to $v_{j,t+1}^{in}, j \neq i$, with same price and capacity.

4. Inter-layer edges capturing leftover bikes from $v_{i,t}$ to $v_{i,t+1}$ are connected from $v_{i,t}^{left}$ to $v_{i,t+1}^{left}$, with same price and capacity.

5. Intra-layer edges capturing relocation by us are connected from $v_{i,t}^{left}$ to $v_{j,t}^{left}$, with same price and capacity.

6. To capture the supply constraints, we add an edge from $v_{i,t}^{in}$ to $v_{i,t}^{left}$ with zero price and infinite capacity.

7. To capture the demand constraints, we add an edge from $v_{i,t}^{left}$ to $v_{i,t}^{out}$ with zero price and infinite capacity.

With these tricks, we still guarantee the dynamics of leftover bikes as in Equation 1.3 are respected, and in the meantime, the additional constraints in the naive model is transformed into constraints on flow on edges as follows:

1. Flow on $(v_{i,t}^{in}, v_{i,t}^{left})$ should be exactly $s_{i,t}$.

2. Flow on $(v_{i,t}^{left}, v_{i,t}^{out})$ should be exactly $d_{i,t}$

The corresponding network diagram is shown in Figure 2.

2.2.1 Proof of correctness

After these tricks, the new model still captures:

1. Constraints:

   (a) The supplying and demanding constraints are captured by flow constraints on $(v_{i,t}^{in}, v_{i,t}^{left})$ and $(v_{i,t}^{left}, v_{i,t}^{out})$, respectively.

   (b) The dynamics of leftover bikes Equation 1.3 is equivalent to the flow conservation property of at each $v_{i,t}^{left}$.

2. Objective: Still, only relocation edges $(v_{i,t}^{left}, v_{j,t}^{left})$ has non-zero price, and the cost of the flow

$$L = \sum_{i}^{n} \sum_{j}^{n} \sum_{t=1}^{Kk} b f_{v_{i,t}^{left}, v_{j,t}^{left}} \tag{2.5}$$

   is the same as Equation 1.5.

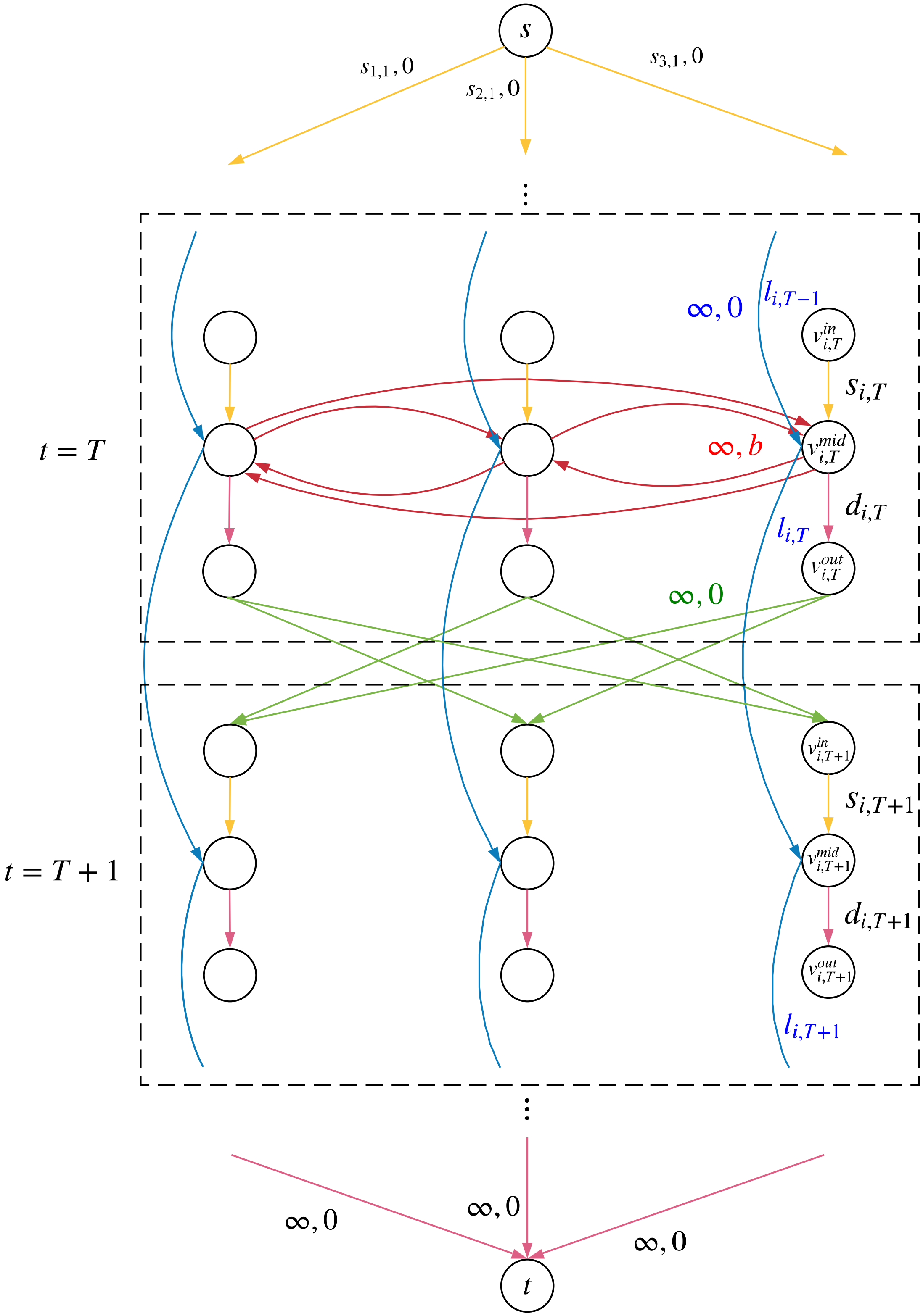

Figure 2: The diagram for the three-node model.

2.3 Reduced Model

Now we need to deal with the exact flow constraints on the edges.

As in the lecture notes, the lower bound of the flow f on an edge (u, v) can be accommodated by adjusting the demand of end nodes with $d'_u = d_u + f$ and $d'_v = d_v - f$. Then, to ensure no more flow are on these edges, we can simply remove the edges in the new graph. For our problem, the tricks are

1. Adjust the demands of nodes as follows: $d_{v_{i,t}^{in}} = s_{i,t}$, $d_{v_{i,t}^{left}} = -s_{i,t} + d_{i,t}$ and $d_{v_{i,t}^{out}} = -d_{i,t}$.

2. Remove edges $(v_{i,t}^{in}, v_{i,t}^{left})$ and $(v_{i,t}^{left}, v_{i,t}^{out})$.

With these tricks, the constraints we develop before for supply and demand are incorporated in the flow network. The mincost flow in this final network corresponds to optimal solution of the bike sharing problem, without any additional constraints. The diagram for the reduced model is shown in Figure 3.

2.3.1 Proof of Correctness

This reduced model is equivalent to our three-node model because the exact flow value constraint is satisfied:

1. The lower bound of each edge on which we require exact flow value is guaranteed with the adjusting demand trick.

2. The upper bound of each edge on which we require exact flow value is guaranteed with removing those edges after lower bound is guaranteed.

2.4 Final Model

To reduce the complexity of our model, we note that in our reduced model, all the inter-layer edges capturing bike movement by riders can be removed, without affecting the solution. After the removal, all the inter-layer nodes $v_{i,t}^{in}, t \neq 1$ and $v_{i,t}^{out}, t \neq Kk$ can also be removed, without affecting solution. The diagram for the final model is shown in Figure 4.

2.4.1 Proof of Correctness

In solving the mincost flow in the reduced model, an artificial source s^* and artificial sink t^* will be connected to all nodes with negative demands (including s) and positive demands (including t) respectively. We then observe that

1. Removing these nodes and edges doesn't affect the flow of rest part of the graph. This is based on the fact that flow from s^* that goes through any node in the set of inter-layer nodes can't go through nodes that is not in this set.

2. A zero-cost flow that satisfies the flow conservation of all inter-layer nodes can be found. This is based on the fact that all the edges connecting inter-layer nodes are with infinite capacity and zero price, and the demands for all inter-layer nodes sum up to zero.

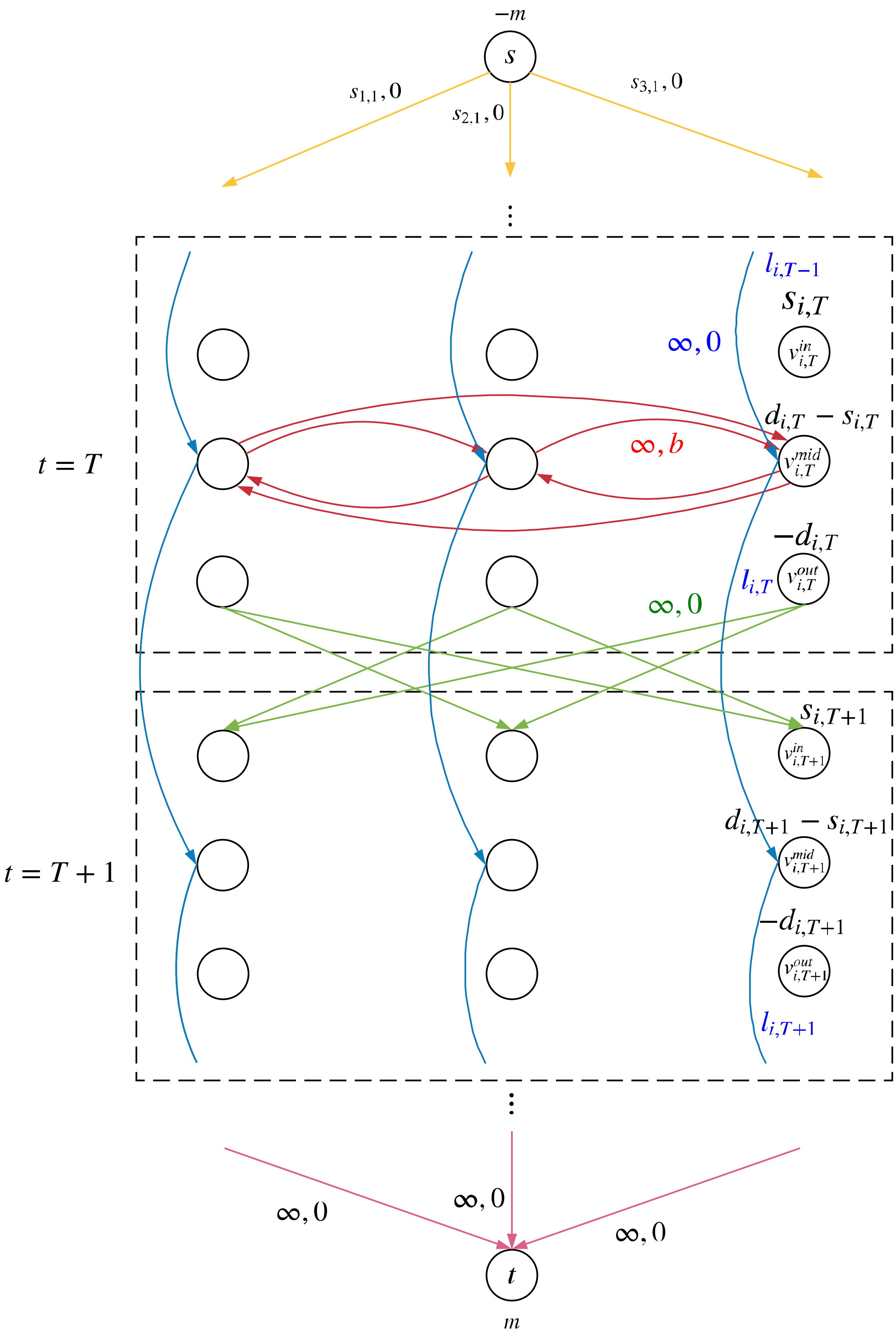

Figure 3: The diagram for the reduced model.

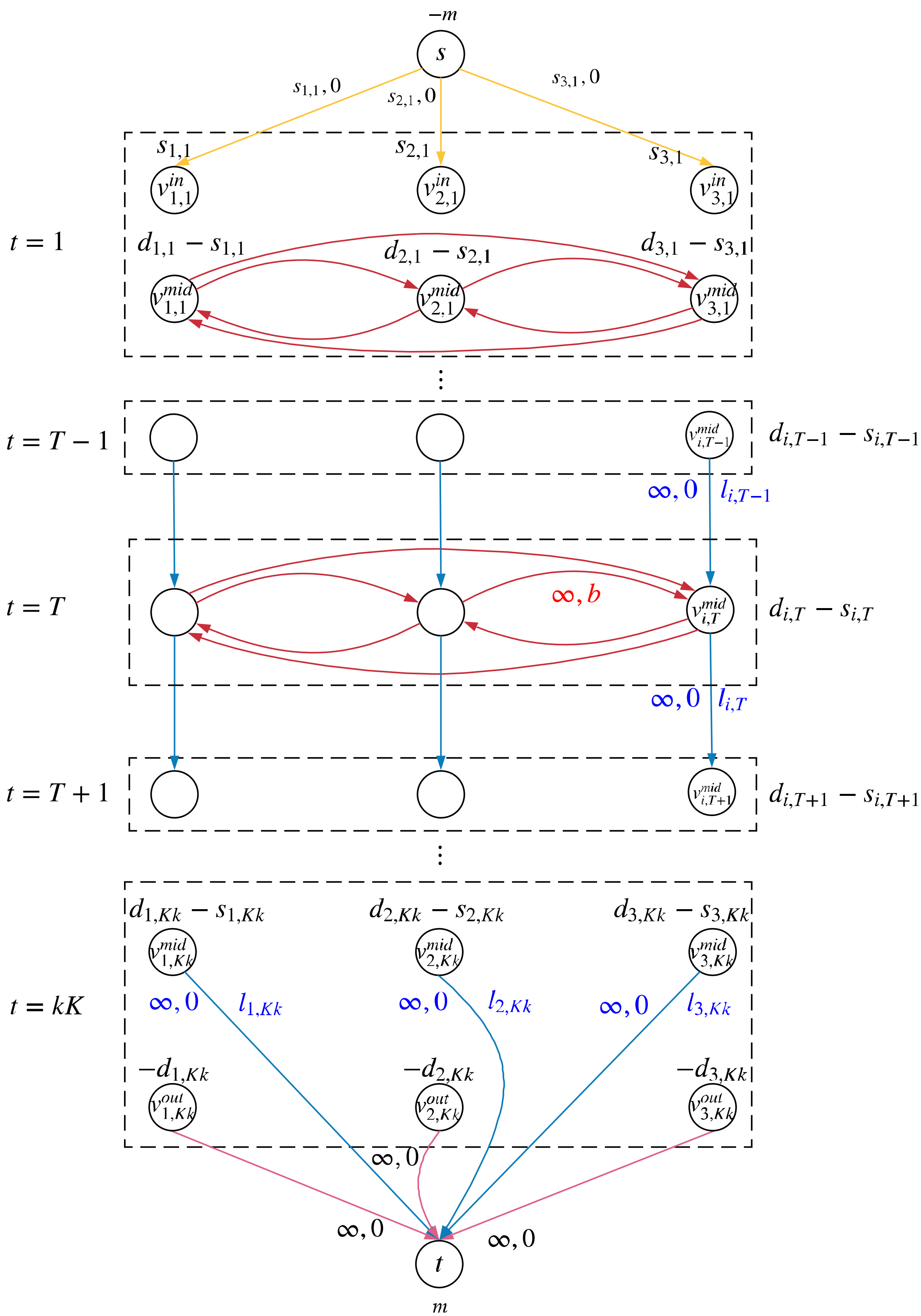

Figure 4: The diagram for the final model.

The first observation tells us removing these nodes and edges doesn't affect the flow conservation constraints and cost of flow of the rest part of the graph. The second observation tells us we can always satisfy the constraints of this part of the graph with zero cost. Combining these we conclude that removing the inter-layer nodes along with edges connecting to them doesn't affect the constraints and objective of the reduced model.

3 Mincost flow solver

Since we reduce the problem as a pure mincost flow problem without any additional constraints, a general mincost flow solver can find the solution.

4 Complexity

The number of nodes in the final model is $|V| = nKk+2n+2$, where 2 represents s and t, $2n$ is the number of nodes in first supply layer and the last demand layer, nKk is the number of nodes of all middle nodes. Number of edges is $|E| = nKk + 2n + 4n(n-1)K$, where $2n$ is the number of edges connecting s and the nodes in the first supply layer and t and the nodes in the last demand layer, nKk is the number of leftover edges and $2 \times 2K \times n(n-1)$ is the number of re-location edges. The current best algorithm for solving mincost flow problem runs in $O(|E|^2 \log |V| + |E|\,|V| \log^2 |V|)$, as in [1].

References

[1] James B Orlin. A faster strongly polynomial minimum cost flow algorithm. *Operations research*, 41(2):338–350, 1993.