BULETINUL ACADEMIEI DE ŞTIINŢE
A REPUBLICII MOLDOVA. MATEMATICA
Number 1(47), 2005, Pages 19–34
ISSN 1024–7696

Optimal multicommodity flows in dynamic networks and algorithms for their finding

M. Fonoberova, D. Lozovanu

Abstract. In this paper we study two basic problems related to dynamic flows: maximum multicommodity flow and the minimum cost multicommodity flow problems. We consider these problems on dynamic networks with time-varying capacities of edges. For minimum cost multicommodity flow problem we assume that cost functions, defined on edges, are nonlinear and depending on time and flow, and the demand function also depends on time. We propose algorithms for solving these dynamic problems, which are based on their reducing to static ones on a time-expanded network.

Mathematics subject classification: 90B10, 90C35, 90C27.
Keywords and phrases: Dynamic networks, dynamic flows, multicommodity flows, maximum flows, minimum cost flows.

1 Introduction

In this paper we study dynamic versions of the maximum multicommodity flow and the nonlinear minimum-cost multicommodity flow problems on networks. These problems generalize the classical static flow problems and extend some dynamic [10, 11] and control models on networks [12]. We propose algorithms for solving these dynamic problems, which are based on their reducing to static ones on a time-expanded network [7]. We also note some different methods for constructing time-expanded networks in the case of acyclic graphs.

For our problems the time is an essential component, either because the flows of some commodity take time to pass from one location to another, or because the structure of network changes over time. Classical static network flow models are known as valuable tools for different applications but they fail to capture the property of many real-life problems. To tackle this problem, we use dynamic network flow models instead of the static ones.

Dynamic flows are widely used to model network-structured, decision-making problems over time: problems in electronic communication, production and distribution, economic planning, cash flow, job scheduling, and transportation [1]) In considered dynamic models the flow passes an arc with time, it can be delayed at nodes, flow values on arcs and the network parameters can change with time. While very efficient solution methods exist for static flow problems, dynamic flow problems have proved to be more difficult to solve.

Dynamic multicommodity flows are among the most important and challenging problems in network optimization, due to the large size of these models in real world applications. Many product distribution, scheduling planning, telecommunication, transportation, communication, and management problems can be formulated and solved as multicommodity flow problems [2]. The multicommodity flow problem consists of shipping several different commodities from their respective sources to their sinks through a given network so that the total flow going through each edge does not exceed its capacity. No commodity ever transforms into another commodity, so that each one has its own flow conservation constraints, but they compete for the resources of the common network. Despite being closer to reality, dynamic multicommodity flow models, because of their complexity, have not been investigated as well as classical ones.

In this paper we study two basic problems related to dynamic flows: maximum multicommodity flow and the minimum cost multicommodity flow problems. We consider these problems on dynamic networks with time-varying capacities of edges. For minimum cost multicommodity flow problem we assume that cost functions, defined on edges, are nonlinear and depending on time and flow. Moreover, we assume that the demand function also depends on time. It is important to notice that if the edge costs do not depend on flow, then the dynamic multicommodity minimum-cost flow problem can be regarded as the network discrete optimal control problem or, equivalently, the problem of finding the shortest paths ([7]) in dynamic networks.

2 Static multicommodity flow problems

In order to study dynamic versions of multicommodity flow problems we will use the following static flow problems.

2.1 The maximum multicommodity flow problem

For the maximum multicommodity flow problem we consider the following static network $N = (V, V_-, V_+, E, K, w, u)$. A flow x on this network assigns every arc $e \in E$ for each commodity $k \in K$ a non-negative flow value x_e^k such that the following flow conservation constraints are obeyed:

$$\sum_{e \in E^+(v)} x_e^k - \sum_{e \in E^-(v)} x_e^k = \begin{cases} -y_v^k, \ v \in V_-^k, \\ 0, \ v \in V_0^k, \\ y_v^k, \ v \in V_+^k, \end{cases} \tag{1}$$

$$y_v^k \geq 0, \ \forall\, v \in V, \ \forall\, k \in K, \tag{2}$$

where $E^+(v) = \{(u,v) \,|\, (u,v) \in E\}$, $E^-(v) = \{(v,u) \,|\, (v,u) \in E\}$, V_-^k, V_+^k and V_0^k are sets of sources, sinks and intermediate nodes for commodity k of network N, $V_- = \cup_{k \in K} V_-^k$, $V_+ = \cup_{k \in K} V_+^k$, $V_0 = \cup_{k \in K} V_0^k$, $V = V_- \cup V_0 \cup V_+$.

The multicommodity flow x is called feasible if it obeys the mutual capacity constraints:

$$\sum_{k\in K} x_e^k \le u_e,\ \forall\, e \in E \tag{3}$$

and individual capacities of every arc for each commodity:

$$0 \le x_e^k \le w_e^k,\ \forall\, e \in E,\ \forall k \in K. \tag{4}$$

These constraints are called weak and strong forcing constraints, respectively.

The maximum multicommodity flow problem consists in maximizing the following objective function:

$$|x| = \sum_{k\in K} \sum_{v\in V_+^k} y_v^k$$

subject to (1)-(4).

2.2 The minimum cost multicommodity flow problem

For the minimum cost multicommodity flow problem we consider the following static network $N = (V, E, K, w, u, d, \varphi)$. A flow x on this network assigns every arc $e \in E$ for each commodity $k \in K = \{1, 2, \dots, k\}$ a non-negative flow value x_e^k such that the following flow conservation constraints are obeyed:

$$\sum_{e\in E^+(v)} x_e^k - \sum_{e\in E^-(v)} x_e^k = d_v^k,\ \forall\, v \in V,\ \forall\, k \in K, \tag{5}$$

where d: $V \times K \to R$ is a demand function and $\sum_{v\in V} d_v^k = 0,\ \forall k \in K$.

The minimum cost multicommodity flow problem consists in minimizing the following objective function:

$$c(x) = \sum_{e\in E} \varphi_e(x_e^1, x_e^2, \dots, x_e^k),$$

subject to (5),(3),(4),

where φ: $E \times R_+ \to R_+$ is a cost function.

The mathematical tool we are going to use for studying and solving dynamical versions of maximum and minimum cost multicommodity flow problems is based on special procedures of their reducing to static problems on auxiliary networks.

3 The dynamic maximum multicommodity flow problem

3.1 The problem formulation

The object of the maximum dynamic flow problem is to send a maximum amount of flow from sources to sinks within a given time bound without violating capacity constraints of any edge. The maximum multicommodity flow problem requires to find the maximum flow of a set of commodities through a network, where the arcs have an individual capacity for each commodity, and a mutual capacity for all the commodities.

We consider the discrete time model, in which all times are integral and bounded by horizon T. The time horizon (finite or infinite) is the time until which the flow can travel in the network and defines the makespan $\mathbb{T} = \{0, 1, \ldots, T\}$ of time moments we consider. Time is measured in discrete steps, so that if one unit of flow leaves node u at time t on arc $e = (u, v)$, one unit of flow arrives at node v at time $t + \tau_e$, where τ_e is the transit time of arc e.

Without loosing generality, we assume that no edges enter sources or exit sinks. We consider that all of the flow is dumped into the network at time 0. Accordingly the sources are nodes through which flow enters the network and the sinks are nodes through which flow leaves the network. The sources and sinks are sometimes called terminal nodes, while the intermediate nodes are called non-terminals. In the case of many sources and sinks the maximum flow problem can be reduced to the standard one by introducing one additional artificial source and one additional artificial sink and edges leading from the new source to true sources and from true sinks to the new sink. The transit times of these new edges are zero and the capacities of edges connecting the artificial source with all other sources are bounded by the capacities of these sources; the capacities of edges connecting all other sinks with the artificial sink are bounded by the capacities of these sinks.

We consider a network $N = (V, V_-, V_+, E, K, w, u, \tau)$ that contains a directed graph $G = (V, E)$ and a set of commodities K that must be routed through the same network. The graph G consists of set of vertices $V = V_- \cup V_+ \cup V_0$, where V_-, V_+ and V_0 are sets of sources, sinks and intermediate nodes, respectively, and set of edges E. Each edge $e \in E$ has a nonnegative time-varying capacity $w_e^k(t)$ which bounds the amount of flow of each commodity $k \in K$ allowed on arc e in every moment of time $t \in \mathbb{T}$. We also consider that every arc $e \in E$ has a nonnegative time-varying capacity for all commodities, which is known as the mutual capacity $u_e(t)$. Moreover, each edge $e \in E$ has an associated positive transit time τ_e which determines the amount of time it takes for flow to travel from the tail to the head of that edge.

A feasible dynamic flow on N is a function x: $E \times K \times \mathbb{T} \to R_+$ that satisfies the following conditions:

$$\sum_{\substack{e \in E^+(v) \\ t-\tau_e \geq 0}} x_e^k(t-\tau_e) - \sum_{e \in E^-(v)} x_e^k(t) = \begin{cases} -y_v^k(t), \ v \in V_-^k, \\ 0, \ v \in V_0^k, \\ y_v^k(t), \ v \in V_+^k, \end{cases} \quad \forall\, t \in \mathbb{T}, \ \forall\, v \in V, \ \forall k \in K; \tag{6}$$

$$y_v^k(t) \geq 0, \ \forall\, v \in V, \forall t \in \mathbb{T}, \ \forall k \in K;$$

$$\sum_{k \in K} x_e^k(t) \leq u_e(t), \ \forall\, t \in \mathbb{T}, \ \forall e \in E; \tag{7}$$

$$0 \leq x_e^k(t) \leq w_e^k(t), \quad \forall\, t \in \mathbb{T}, \ \forall\, e \in E, \ \forall k \in K; \tag{8}$$

$$x_e^k(t) = 0, \ \forall\, e \in E, \ t = \overline{T - \tau_e + 1, T}, \ \forall k \in K. \tag{9}$$

Here the function x defines the value $x_e^k(t)$ of flow of commodity k entering edge e at time t. It is easy to observe that the flow of commodity k does not enter edge e at time t if it will have to leave the edge after time T; this is ensured by condition (9). Capacity constraints (8) mean that in a feasible dynamic flow, at most $w_e^k(t)$ units of flow of commodity k can enter arc e at time t. Mutual capacity constraints (7) mean that in a feasible dynamic flow, at most $u_e(t)$ units of flow can enter arc e at time t. Conditions (6) represent flow conservation constraints.

The value of the dynamic flow x is defined as follows:

$$|x| = \sum_{k \in K} \sum_{t \in \mathbb{T}} \sum_{v \in V_+^k} y_v^k(t).$$

The object of the maximum multicommodity flow problem is to find a feasible flow that maximizes this objective function.

It is easy to observe that if $\tau_e = 0, \forall\, e \in E$ and $T = 0$ then the formulated problem becomes the static multicommodity flow problem.

3.2 The main results

In this paper we propose an approach for solving the formulated problem, which is based on reduction of this problem to a static well studied one. We show that the maximum multicommodity flow problem on network N can be reduced to a static problem on an auxiliary network N^T; we name this network as a time-expanded network. The advantage of this approach is that it turns the problem of determining a maximum dynamic flow into a classical static maximum flow problem on the time-expanded network.

The time-expanded network is a static representation of the dynamic network. The essence of the time-expanded network is that it contains a copy of the vertices of the dynamic network for each moment of time, and the transit times and flows are implicit in the edges linking those copies. In such a way, a dynamic flow problem in a given network with transit times on the arcs can be transformed into an equivalent static flow problem in the corresponding time-expanded network. A discrete dynamic flow in the given network can be interpreted as a static flow in the corresponding time-expanded network.

We define the time-expanded network as follows:

1. $V^T\colon\ = \{v(t)\,|\,v \in V,\ t \in \mathbb{T}\}$;
2. $E^T\colon\ = \{e(t) = (v(t), w(t+\tau_e))\,|\,e = (v,w) \in E,\ 0 \le t \le T - \tau_e\}$;
3. $u^T_{e(t)}\colon\ = u_e(t)$ for $e(t) \in E^T$;
4. ${w^k_{e(t)}}^T\colon\ = w^k_e(t)$ for $e(t) \in E^T,\ k \in K$.

Let $e(t) = (v(t), w(t+\tau_e)) \in E^T$ and let $x^k_e(t)$ be a flow of commodity $k \in K$ on the dynamic network N. The corresponding function on the time-expanded network N^T is defined as follows:

$$ {x^k_{e(t)}}^T = x^k_e(t),\ \forall k \in K. \tag{10} $$

Lemma 1. *The correspondence (10) is a bijection from the set of feasible flows on the dynamic network N onto the set of feasible flows on the time-expanded network N^T.*

Proof. It is obvious that the correspondence above is a bijection from the set of T-horizon functions on the dynamic network N onto the set of functions on the time-expanded network N^T. It is easy to observe that a feasible flow on the dynamic network N is a feasible flow on the time-expanded network N^T and vice-versa. Indeed, individual capacity constraints are obeyed:

$$ 0 \le {x^k_{e(t)}}^T = x^k_e(t) \le w^k_e(t) = {w^k_{e(t)}}^T,\ \forall\, e \in E,\ \ \forall\, t \in \mathbb{T},\ \forall\, k \in K $$

and mutual capacity constraints are also obeyed:

$$ \sum_{k \in K} {x^k_{e(t)}}^T = \sum_{k \in K} x^k_e(t) \le u_e(t) = u^T_{e(t)},\ \forall\, t \in \mathbb{T},\ \forall\, e \in E. $$

Therefore it is enough to show that each dynamic flow on the dynamic network N is put into the correspondence with a static flow on the time-expanded network N^T and vice-versa.

Henceforward we define

$$ d^k_v(t) = \begin{cases} -y^k_v(t),\ v \in V^k_-, \\ 0,\ v \in V^k_0, \\ y^k_v(t),\ v \in V^k_+, \end{cases} \qquad y^k_v(t) \ge 0,\ \forall v \in V,\ \forall t \in \mathbb{T},\ \forall k \in K. $$

Let $x_e^k(t)$ be a dynamic flow of commodity k on N and let ${x_{e(t)}^k}^T$ be a corresponding function on N^T. Let's prove that ${x_{e(t)}^k}^T$ satisfies the conservation constraints on its static network. Let $v \in V$ be an arbitrary node in N and $t \in \mathbb{T}$ an arbitrary moment of time:

$$d_v^k(t) \overset{(i)}{=} \sum_{\substack{e \in E^+(v) \\ t-\tau_e \geq 0}} x_e^k(t-\tau_e) - \sum_{e \in E^-(v)} x_e^k(t) =$$

$$= \sum_{e(t-\tau_e) \in E^+(v(t))} {x_{e(t-\tau_e)}^k}^T - \sum_{e(t) \in E^-(v(t))} {x_{e(t)}^k}^T \overset{(ii)}{=} {d_{v(t)}^k}^T. \tag{11}$$

Note that according to the definition of the time-expanded network the set of edges $\{e(t-\tau_e) | e(t-\tau_e) \in E^+(v(t))\}$ consists of all edges that enter $v(t)$, while the set of edges $\{e(t) | e(t) \in E^-(v(t))\}$ consists of all edges that originate from $v(t)$. Therefore, all necessary conditions are satisfied for each node $v(t) \in V^T$. Hence, ${x_{e(t)}^k}^T$ is a flow on the time-expanded network N^T.

Let ${x_{e(t)}^k}^T$ be a static flow of commodity k on the time-expanded network N^T and let $x_e^k(t)$ be a corresponding function on the dynamic network N. Let $v(t) \in V^T$ be an arbitrary node in N^T. The conservation constraints for this node in the static network are expressed by equality (ii) from (11), which holds for all $v(t) \in V^T$ at all times $t \in \mathbb{T}$. Therefore, equality (i) holds for all $v \in V$ at all times $t \in \mathbb{T}$ and $x_e^k(t)$ is a flow on the dynamic network N. $\square$

In the following lemma we prove that values of any dynamic multicommodity flow and corresponding static multicommodity flow in the time-expanded network are equal.

Lemma 2. *If x is a flow on the dynamic network N and x^T is a corresponding flow on the time-expanded network N^T, then*

$$|x| = |x^T|.$$

Proof. The proof is straightforward:

$$|x| = \sum_{k \in K} \sum_{t \in \mathbb{T}} \sum_{v \in V_+^k} y_v^k(t) = \sum_{k \in K} \sum_{t \in \mathbb{T}} \sum_{v(t) \in {V_+^k}^T} {y_{v(t)}^k}^T = |x^T|. \qquad \square$$

The above lemmas imply the validity of the following theorem:

Theorem 3. *For each maximum multicommodity flow in the dynamic network there is a corresponding maximum multicommodity flow in the static time-expanded network.*

In such a way, we can solve the considered problem by reducing it to the maximum static multicommodity flow problem, solving the obtained problem in the corresponding time-expanded network and then reconstructing the solution to the solution of the initial problem. Therefore, the maximum multicommodity flow problem on dynamic networks can be solved by applying network flow optimization techniques for static flows directly to the expanded network.

3.3 The algorithm

Let the dynamic network N be given. Our object is to solve the maximum multicommodity flow problem on N. Proceedings are following:

1. Building the time-expanded network N^T for the given dynamic network N.
2. Solving the classical maximum multicommodity flow problem on the static network N^T, using one of the known algorithms [6, 8, 9, 13].
3. Reconstructing the solution of the static problem on N^T to the dynamic problem on N. □

3.4 Some special cases

3.4.1 The case of limited time

In the above items we have discussed the problem of determining the maximum dynamic multicommodity flow from the zero time moment to the fixed time horizon T. In such problems we find the maximum amount of flow until the time T. In many practical cases it is necessary to know the maximum flow in the time period from t_1 to t_2, where $t_1 < t_2$. To obtain the solution of of this problem we have to construct a time-expanded network, the discrete time moments of which form the following makespan $\mathbb{T} = \{t_1, t_1 + 1, \ldots, t_2 - 1, t_2\}$. In that way, by constructing such a time-expanding network and finding the maximum flow in this network we can obtain the maximum flow in the dynamic network for the time period from t_1 to t_2.

3.4.2 The case of two-sided restrictions

The same argumentation as in the above items can be held to solve the maximum multicommodity flow problem on the dynamic networks in the case when, instead of the condition (8) in the definition of the feasible dynamic flow, the following condition takes place:

$$r_e^k(t) \leq x_e^k(t) \leq \overline{r}_e^k(t), \quad \forall\, t \in \mathbb{T},\ \forall\, e \in E,\ \forall\, k \in K$$

where $r_e^k(t)$ and $\overline{r}_e^k(t)$ are lower and upper boundaries of the capacity of the edge e correspondingly. In this case we introduce one additional artificial source b_1 and one additional artificial sink b_2. For every arc $e = (u, v)$, where $r_e^k(t) \neq 0$ we introduce arcs (b_1, v) and (u, b_2) with $r^k(t)$ and 0 as the upper and lower boundaries of the capacity of the edges. We reduce $\overline{r}^k(t)$ to $\overline{r}^k(t) - r^k(t)$, but $r^k(t)$ to 0. We also introduce the arc (b_2, b_1) with $\overline{r}^k_{(b_2,b_1)} = \infty$ and $r^k_{(b_2,b_1)} = 0$. The transit times of all introduced arcs are zero. In such a mode we obtain a new network, on which we can solve the standard maximum flow problem.

4 The dynamic minimum cost multicommodity flow problem

4.1 The problem formulation

The minimum cost flow problem is the problem of sending flows in a network from supply nodes to demand nodes at minimum total cost such that link capacities are not exceeded. The minimum cost multicommodity dynamic flow problem asks to find the flow of a set of commodities through a network with given time horizon, satisfying all supplies and demands with minimum cost.

As in the chapter 3 we consider the discrete time model, in which all times are integral and bounded by horizon T. Time is measured in discrete steps, the set of time moments we consider is $\mathbb{T} = \{0, 1, \dots, T\}$.

We consider a directed network $N = (V, E, K, w, u, \tau, d, \varphi)$ with set of vertices V, set of edges E and set of commodities K that must be routed through the same network. A dynamic network N consists of capacity function w: $E \times K \times \mathbb{T} \to R_+$, mutual capacity function u: $E \times \mathbb{T} \to R_+$, transit time function τ: $E \to R_+$, demand function d: $V \times K \times \mathbb{T} \to R$ and cost function φ: $E \times R_+ \times \mathbb{T} \to R_+$. The demand function $d_v^k(t)$ satisfies the following conditions:

a) there exists $v \in V$ for every $k \in K$ with $d_v^k(0) < 0$;

b) if $d_v^k(t) < 0$ for a node $v \in V$ for commodity $k \in K$ then $d_v^k(t) = 0$, $t = 1, 2, \dots, T$;

c) $\sum\limits_{t \in \mathbb{T}} \sum\limits_{v \in V} d_v^k(t) = 0, \forall k \in K$.

A feasible dynamic flow on N is a function x: $E \times K \times \mathbb{T} \to R_+$ that satisfies conditions (7)-(9) and the following conditions:

$$\sum_{\substack{e \in E^+(v) \\ t - \tau_e \geq 0}} x_e^k(t - \tau_e) - \sum_{e \in E^-(v)} x_e^k(t) = d_v^k(t), \ \forall\, t \in \mathbb{T}, \ \forall\, v \in V, \ \forall k \in K.$$

To model transit costs, which may change over time, we define the cost function $\varphi_e(x_e^1(t), x_e^2(t), \dots, x_e^k(t), t)$ which indicates the cost of shipping flows over edge e entering the edge e at time t.

The total cost of the dynamic multicommodity flow x is defined as follows:

$$c(x) = \sum_{t \in \mathbb{T}} \sum_{e \in E} \varphi_e(x_e^1(t), x_e^2(t), \dots, x_e^k(t), t).$$

The object of the minimum cost multicommodity flow problem is to find a feasible flow that minimizes this objective function.

It is important to notice that in many practical cases cost functions are presented in the following form:

$$\varphi_e(x_e^1(t), x_e^2(t), \dots, x_e^k(t), t) = \sum_{k \in K} \varphi_e^k(x_e^k(t), t).$$

The separable case of cost functions represents the most important one from the practical standpoint. In the case when $\varphi_e^k(x_e^k(t), t)$ are linear functions the dynamic version of the considered problem is reduced to static linear programming problem on an auxiliary static network.

It is easy to observe that if $\tau_e = 0, \forall e \in E$ and $T = 0$ then the formulated problem becomes the static minimum cost multicommodity flow problem.

4.2 The main results

To solve the minimum cost multicommodity flow problem by its reduction to a static one we define the time-expanded network N^T as follows:

1. $V^T \colon = \{v(t) \,|\, v \in V,\ t \in \mathbb{T}\}$;
2. $E^T \colon = \{e(t) = (v(t), w(t + \tau_e)) \,|\, e = (v, w) \in E,\ 0 \le t \le T - \tau_e\}$;
3. $u_{e(t)}^T \colon = u_e(t)$ for $e(t) \in E^T$;
4. ${w_{e(t)}^k}^T \colon = w_e^k(t)$ for $e(t) \in E^T,\ k \in K$.
5. $\varphi_{e(t)}^T({x_{e(t)}^1}^T, {x_{e(t)}^2}^T, \dots, {x_{e(t)}^k}^T) \colon = \varphi_e(x_e^1(t), x_e^2(t), \dots, x_e^k(t), t)$ for $e(t) \in E^T$, $k \in K$;
6. ${d_{v(t)}^k}^T \colon = d_v^k(t)$ for $v(t) \in V^T,\ k \in K$.

If we define a flow correspondence by relation (10), it can be proved, using the same method as in Lemma 1, that the set of feasible flows on the dynamic network N corresponds to the set of feasible flows on the time-expanded network N^T.

In the following lemma we prove that costs of any dynamic multicommodity flow and corresponding static multicommodity flow in the time-expanded network are equal.

Lemma 4. *If x is a flow on the dynamic network N and x^T is a corresponding flow on the time-expanded network N^T, then*

$$c(x) = c^T(x^T).$$

Proof. The proof is straightforward:

$$c(x) = \sum_{t\in\mathbb{T}} \sum_{e\in E} \varphi_e(x_e^1(t), x_e^2(t), \ldots, x_e^k(t), t) =$$

$$= \sum_{t\in\mathbb{T}} \sum_{e(t)\in E^T} \varphi^T_{e(t)}({x^1_{e(t)}}^T, {x^2_{e(t)}}^T, \ldots, {x^k_{e(t)}}^T) = c^T(x^T). \qquad \square$$

Using the above results we obtain the following theorem:

Theorem 5. *For each minimum-cost multicommodity flow in the dynamic network there is a corresponding minimum-cost multicommodity flow in the static time-expanded network.*

In such a way, the minimum cost multicommodity flow problem on the dynamic network can be solved by static flow computations in the corresponding time-expanded network. The solution of the considered problem can be obtained by using the solution of the static minimum cost multicommodity flow problem on the time-expanded network.

4.3 The algorithm

Let the dynamic network N be given. The minimum-cost multicommodity flow problem is to be solved on N. Proceedings are following:

1. Building the time-expanded network N^T for the given dynamic network N.

2. Solving the classical minimum-cost multicommodity flow problem on the static network N^T ([3–6]).

3. Reconstructing the solution of the static problem on N^T to the dynamic problem on N. $\square$

4.4 Generalization

Now let us study some general cases of the minimum cost dynamic multicommodity flow problems. First of all, we assume that only a part of the flow is dumped into the considered network at the time 0, i.e. the condition b) in the definition of the demand function $d_v^k(t)$ doesn't hold. Using the following, this case can be reduced to the one considered above.

Let the flow is dumped into the network from the node $v \in V$ at an arbitrary moment of time $\mathtt{t}$, different from the ordinary moment. We can reduce this problem to the problem, in which all of the flow is dumped into the network at the initial time by introducing loops in all nodes from V, except the node v, from which the flow is dumped into the network at the time $\mathtt{t}$. For such loops we attribute transit times which are equal to the time t. So, we can consider that all the flow is dumped in the network at the time t, which we define as the initial time.

The argumentation is the same, when the flow is dumped in the network from different nodes at different moments of time. Let t be the maximum of those moments. In this case we take t as the initial time and construct loops from all the nodes, except those that dump the flow in the network at time t. The transit times of these loops are equal to the difference between time t and the time when the flow from those nodes that generate loops is dumped in the network. So, we reduce this problem to the one, considered above, where the whole flow is dumped into the network at the initial moment of time.

Further we consider the variation of the dynamic network when the condition c) in the definition of the demand function $d_v^k(t)$ doesn't hold. We assume that after time $t = T$ there still is flow in the network, i.e. the following condition is true:

$$\sum_{t \in \mathbb{T}} \sum_{v \in V} d_v^k(t) \leq 0.$$

We can reduce this problem to the problem without flow in the network after an upper bound of time by introduction of the additional node v and additional edges. The rest of the flow in the network is sent to the node v through the arcs, which we just introduced. In such a way we obtain the initial model of the dynamic network.

The next model of the dynamic network is the one when we allow flow storage at the nodes. In this case we can reduce this dynamic network to the initial one by introducing the loops in those nodes in which there is flow storage. The flow which was stored at the nodes passes through these loops. Accordingly, we reduce this problem to the initial one.

The other variation of the dynamic network is the one when the cost functions also depend on the flow at the nodes. In this case we can reduce this model of the dynamic network to the initial one by introducing loops and attributing the cost functions, which were defined in the nodes, to these loops. Consequently, we obtain the initial model of the dynamic network.

The same reasoning to solve the minimum cost flow problem on the dynamic networks and its generalization can be held in the case when, instead of the condition (8) in the definition of the feasible dynamic flow, the following condition takes place:

$$r_e^k(t) \leq x_e^k(t) \leq \overline{r}_e^k(t), \quad \forall\, t \in \mathbb{T},\ \forall\, e \in E,\ \forall\, k \in k$$

where $r_e^k(t)$ and $\overline{r}_e^k(t)$ are lower and upper boundaries of the capacity of the edge e correspondingly.

5 The time-expanded network in the case of acyclic graphs

We will consider the dynamic network N, where the graph $G = (V, E)$ does not contain directed cycles. Let $T^* = \max\{|L|\} = \max\{\sum_{e \in L} \tau_e\}$, where L is a directed path in the graph G. In [10] it is shown that $x_e^k(t) = 0$ for $e \in E$, $k \in K$, $t \geq T^*$.

Using this result we can construct the time expanded-network that consists of $n(T^*+1)$ nodes and $m(T^*+1)$ edges, where n and m are numbers of nodes and edges in the initial network. Since the maximum number of edges a directed path can have in an acyclic network is $n-1$, it immediately results that the time-expanded network has not more than n^2 nodes and mn edges. In such a way, we have established a polynomial upper bound for the size of the time-expanded network.

It is easy to note that in many cases the large majority of intermediate nodes are not connected with a directed path both to a sink and a source. Removing such nodes from the considered network does not influence the set of flows on this network. We will call these nodes irrelevant to the flow problem. Intermediate nodes that are not irrelevant will be denoted relevant. The static network obtained by eliminating the irrelevant nodes and all edges adjacent to them from the time-expanded network will be called the reduced time-expanded network.

We propose the following algorithm for constructing the reduced network based on the process of elimination of irrelevant nodes from the time-expanded network.

Algorithm

1. Building the time-expanded network N^{T^*} for the given dynamic network N.

2. Performing a breadth-first parse of the nodes for each source from the time expanded-network. The result of this step is the set $V_-(V_-^{T^*})$ of the nodes that can be reached from at least a source in V^{T^*}.

3. Performing a breadth-first parse of the nodes beginning with the sink for each sink and parsing the edges in the direction opposite to their normal orientation. The result of this step is the set $V_+(V_+^{T^*})$ of nodes from which at least a sink in V^{T^*} can be reached.

4. The reduced network will consist of a subset of nodes V^{T^*} and edges from E^{T^*} determined in the following way

$$V'^{T^*} = V^{T^*} \cap V_-(V_-^{T^*}) \cap V_+(V_+^{T^*}), \qquad E'^{T^*} = E^{T^*} \cap (V'^{T^*} \times V'^{T^*}).$$

5. ${d'}_{v(t)}^{k\;T^*} := d_v^k(t)$ for $v(t) \in V'^{T^*},\ k \in K$.

6. ${u'}_{e(t)}^{T^*} := u_e(t)$ for $e(t) \in E'^{T^*}$.

7. ${w'}_{e(t)}^{k\;T^*} := w_e^k(t)$ for $e(t) \in E'^{T^*},\ k \in K$.

8. $\varphi_{e(t)}^T({x^1_{e(t)}}^T, {x^2_{e(t)}}^T, \ldots, {x^k_{e(t)}}^T) := \varphi_e(x_e^1(t), x_e^2(t), \ldots, x_e^k(t), t)$ for $e(t) \in E^T$, $k \in K$. $\square$

The complexity of this algorithm can be estimated to be the same as the complexity of constructing the time-expanded network. In [10] it is proved that the reduced network can be used in place of the time-expanded network. This algorithm begins with the dynamic network containing a small number of nodes, builds the time-expanded network with the largest number of nodes and then selects from it the reduced network with a smaller number of nodes.

Now we propose an algorithm for constructing the reduced network N'^{T^*} directly from the dynamic network N.

Algorithm

1. Building the dynamic network N', which contains all the nodes in N except those that are not connected with a direct path with at least a sink and at least a source, employing the same method as used in the above algorithm for static networks.

2. Creating queue $C = \{v_1(0), v_2(0), \dots, v_l(0)\}$, where $\{v_1, v_2, \dots, v_l\} = V_-$. We consider only $v_i(0)$, $v_i \in V_-$, because all of the flow is dumped into the network at time 0.

3. Initializing sets:

$$V_-'^{T^*} = \emptyset, \qquad V_+'^{T^*} = \{v_i(t) | v_i \in V_+, t \in \mathbb{T}\}, \qquad V'^{T^*} = V_-'^{T^*} \cup V_+'^{T^*}.$$

4. While queue C is not empty execute for each node $v_1(t_1)$ at the head of the queue:

   a) If node $v_1(t_1)$ is already in V'^{T^*}, then jump to step (4d).

   b) For each $(v_1, v_i) \in E^-(v_1)$ in the dynamic network execute:

   - **i)** If node $v_i \in V_0$ and node $v_i(t_1 + \tau_{(v_1,v_i)})$ is not already in V'^{T^*} then add node $v_i(t_1 + \tau_{(v_1,v_i)})$ to queue C and add edge $(v_1(t_1), v_i(t_1 + \tau_{(v_1,v_i)}))$ to E'^{T^*}.
   - **ii)** If node $v_i \in V_+$ and edge $(v_1(t_1), v_i(t_1 + \tau_{(v_1,v_i)}))$ is not already in E'^{T^*}, then add edge $(v_1(t_1), v_i(t_1 + \tau_{(v_1,v_i)}))$ to E'^{T^*}.

   c) Add node $v_1(t_1)$ to V'^{T^*}.

   d) Remove node $v_1(t_1)$ from queue C, all nodes moving one step closer to the head of the queue.

5. ${d'^k_{v(t)}}^{T^*} := d^k_v(t)$ for $v(t) \in V'^{T^*}$, $k \in K$.

6. ${u'_{e(t)}}^{T^*} := u_e(t)$ for $e(t) \in E'^{T^*}$.

7. ${w'^k_{e(t)}}^{T^*} := w^k_e(t)$ for $e(t) \in E'^{T^*}$, $k \in K$.

8. $\varphi^T_{e(t)}({x^1_{e(t)}}^T, {x^2_{e(t)}}^T, \ldots, {x^k_{e(t)}}^T) := \varphi_e(x^1_e(t), x^2_e(t), \ldots, x^k_e(t), t)$ for $e(t) \in E^T$, $k \in K$. □

The network $N^{'T^*}$ built by this algorithm contains only intermediate nodes from the time-expanded network N^{T^*} that are relevant. Furthermore it contains all intermediate nodes with this property from N^{T^*}. The functions d, u, w, φ are the same as those on the reduced network and the built network is the reduced network.

6 Conclusions

In this paper we formulated and studied the maximum and minimum cost multicommodity dynamic flow problems on dynamic networks with time-varying capacities of edges. For minimum cost multicommodity flow problem we assumed that cost functions, defined on edges, are nonlinear and depending on time and flow, and the demand function also depends on time. To solve the proposed problems we reduced them to the static ones on auxiliary networks and proposed corresponding algorithms.

At the end we would like to note that the same argumentation and algorithms as were described for the multicommodity flow problems are evidently hold for one-commodity flow problems, one-commodity flow being a particular case of multicommodity flows. The difference consists in the fact that for one-commodity flow instead of individual and mutual capacity constraints only individual capacity constraints are considered. One-commodity flow can be regarded as multicommodity flow in the case of only one commodity.

References

[1] Aronson J. *A survey of dynamic network flows.* Ann. Oper. Res., 1989, **20**, p. 1–66.

[2] Assad A. *Multicommodity network flows: A survey.* Networks, 1978, **8**, p. 37–92.

[3] Castro J., Nabona N. *An implementation of linear and nonlinear multicommodity network flows.* European Journal of Operational Research Theory and Methodology, 1996, **92**, p. 37–53.

[4] Castro J. *A specialized interior-point algorithm for multicommodity network flows.* Siam J. Optim., 2000, N 10(3), p. 852—877.

[5] Castro J. *Solving Difficult Multicommodity Problems with a Specialized Interior-Point Algorithm.* Annals of Operations Research, 2003, **124**, p. 35-–48.

[6] Fleisher L. *Approximating multicommodity flow independent of the number of commodities.* Siam J. Discrete Math., 2000, N 13(4), p. 505-–520.

[7] Ford L., Fulkerson D. *Flows in Networks.* Princeton University Press, Princeton, NJ, 1962.

[8] Goldberg A., Tarjan R. *A New Approach to the Maximum-Flow Problem.* Journal of the Association for Computing Machinery, 1988, **35**, N 4, p. 921–940.

[9] Kumar S., Gupta P. *An Incremental Algorithm for the Maximum Flow Problem.* Journal of Mathematical Modelling and Algorithms, 2003, **2**, p. 1—16.

[10] Lozovanu D., Stratila D. *The minimum-cost flow problem on dynamic networks and algorithm for its solving*. Bul. Acad. Ştiinţe Repub. Mold., Matematica, 2001, N 3(37), p. 38–56.

[11] Lozovanu D., Stratila D. *Optimal flow in dynamic networks with nonlinear cost functions on edges. Analysis and optimization of differential systems.* (Edited by V. Barbu and I. Lesiencko), ISBN 1–4020–7439–5. Klnwer Academic Publissers, 2003, p. 247–258.

[12] Lozovanu D., Pickl S. *A special dynamic programming technique for multiobjective discrete control and for dynamic games on graph-based methods*. CTWO4 Workshop on Graphs and Combinatorial Optimization, Milano, 2004, p. 184–188.

[13] Mazzoni G., Pallottino S., Scutella M. *The maximum flow problem: A max-preflow approach*. European Journal of Operational Research, 1991, **53**, p. 257–278.

Institute of Mathematics and Computer Science
5 Academiei str.
Chişinău, MD−2028, Moldova
E-mails: *mashaf83@yahoo.com*, *lozovanu@math.md*

Received March 14, 2005