# Optimal Flow Routing in Multi-hop Sensor Networks with Real-Time Constraints through Linear Programming.

Jiří Trdlička
CTU Faculty of Electrical Engineering
Department of Control Engineering
Karlovo nám. 13, Prague 2, Czech Republic
trdlij1@fel.cvut.cz

Zdeněk Hanzálek
CTU Faculty of Electrical Engineering
Department of Control Engineering
Karlovo nám. 13, Prague 2, Czech Republic
hanzalek@fel.cvut.cz

Mikael Johansson
KTH School of Electrical Engineering
Automatic Control
Osquldas väg 10, SE-100 44 Stockholm, Sweden
mikael.johansson@s3.kth.se

## Abstract

*We have proposed an algorithm for optimal real-time routing in multi-hop communication networks for multi-source/multi-sink connection. The algorithm deals with various capacity constraints in terms of communication limits and real-time constraints expressed as deadline for each particular flow of data. The objective is to find the optimal routing in terms of energy consumption. The algorithm is based on a data flow model leading to Linear Programming formulation and therefore it ensures polynomial-time complexity. An extension handling simultaneous real-time and non real-time routing is added. An example of data collection from 100 nodes is presented and performance experiments illustrating time complexity in dependence on the number of nodes are given.*

## 1. Introduction

Our work is focused on data flow routing through the multi-hop static network, where all data has to be delivered to the destinations in time. An example of a target application could be a network for periodic sensing and control of some commodity consumption (like electrical energy consumption, gas consumption, water consumption, etc.) in large objects, like airports, supermarkets, etc. Then each sensing device produces a data flow of a particular volume, which is supposed to be routed through the network. We optimize the energy consumption for data transfer and we assume the following constraints: link capacities, node capacities and different deadlines for each sensed value. The solution where not all data is delivered before the deadlines is not feasible. We assume a TDMA-like medium access protocol (e.g. GTS allocation in IEEE 802.15.4 [11, 12]) which ensures collision-free communication and causes communication delay. Due to the TDMA mechanism assumed, the worst case delay from the source node to the destination node is a sum of the particular delays for each of the hops, assumed to be an integer (derived from the parameters like TDMA period, worst case execution time of the communication stack...). In a particular setting, we may assume a unit delay common for all hops (the same TDMA period, negligable influence of the transmission delay on physical layer...). Therefore, we assume the unit hop delay (the deadlines are expressed as the number of communication hops between devices) and further more we generalize our approach to integer delays that may differ for each hop [16]. Assuming only the communication delay caused by the TDMA (no execution time of the communication stack...), the hop delay would be shorter than the TDMA period, depending on the routing direction and on the TDMA schedule [13], at the cost of the model's complexity.

The optimal routing of data flow depends on the link capacities and on the communication prices (e.g. energy consumption). We take the value of the data flow as a continuous quantity and we allow the flow fragmentation go to more routing paths. Thanks to the flow continuity, the problem is solvable in polynomial time by Linear Programming. The network topology is represented by a directed graph where the nodes represent the devices and the oriented edges represent the oriented communication links between the devices.

Traditionally, routing problems for data networks have often been formulated as linear or convex multicommodity network flow routing problems (e.g. [4, 15]) for which many efficient solution methods exist [3, 14, 5, 10].

In [19], the multicommodity problem formulation is used for simultaneous routing and resource allocation (e.g. node related bandwith), which allowed us to find more efficient routing paths than the paths that would be found in the case of separated flow routing and resource allocation. One of the advantages of the multicommodity network flow model based on convex optimization is that several constraints can be put together. Using the same underlying network model, we can easily combine the solution presented in this article (network routing with real-time constraints) with [19] (network routing and resource allocation). Unfortunately an explanation of such a model is rather complex and therefore, we have not presented resource allocation in this paper in order to simplify the presentation.

Several papers have been performed in the area of real-time routing in wireless sensor networks [2, 1]. In [1], the theoretical work about the capacity of the real-time communication limits in multi-hop wireless sensor networks is presented. In [9], the soft real-time communication protocol in multi-hop wireless sensor networks, called SPEED, is presented. The protocol uses the speed of message propagation to set priorities of the messages. In [7], the protocol called RPAR is presented. It uses the fact that the message propagation speed depends on the transmitting energy and this fact is used to set the priority and transmitting energy according to the remaining time.

According to our current knowledge, there are at least two other papers about treating specific problems of real-time routing in wireless sensor networks. In [6], the authors assume nodes in hexagonal cells and use inter-cell and intra-cell communication in single directions to ensure the real-time behavior. The second protocol presented in [18] uses the distance from the last transmitting node to avoid data collisions and the data is sent in communication waves. However, none of these algorithms can ensure real-time and energy optimal routing.

The multicommodity network flow model is used, because it does not need any particular network structure, like a hexagonal structure (in [6]), or tree topology. This approach can handle any network topology with any number of sources and destinations. In contrast with all papers about real-time routing in sensor networks, referenced in this article, our approach ensures the real-time and energy optimal routing for all communication demands even in high loaded networks.

The paper is organized as follows: Section 2 describes the network structure, the multicommodity network flow model for data flow routing and the formulation of the objective function in terms of energy consumption. In Section 3, which is the main part of our work, our model for real-time data flow routing is presented and an intuitive meaning of our approach is illustrated. We have also extended the approach to simultaneous real-time and non real-time data flow routing and deal with time complexity of the algorithm in this section. An example with 100 nodes and time complexity experiments is given in Section 4. Section 5 concludes the article and mentions the potential future work.

## 2. Multicommodity network flow model

Several papers have been written about multicommodity network flow routing. In this section, we briefly summarize the basic terminology and specify the multicommodity network flow model used in this paper.

### 2.1. Network structure

The network is represented by a directed graph, where for each device able to send or receive data, a node of the graph exists. The nodes are labelled as $n = 1, \ldots, N$. Directed communication links are represented as ordered pairs $(i, j)$ of distinct nodes. The presence of a link $(i, j)$ means that the directed communication, from node $i$ to node $j$, is possible. The links are labelled as $l = 1, \ldots, L$. We define the set of the links $l$ leaving the node $n$ as $\mathcal{O}(n)$ and the set of the links $l$ incoming to node $n$ as $\mathcal{I}(n)$. Each link is only in one set $\mathcal{O}(n)$ of some node $n$ and only in one set $\mathcal{I}(n)$ of some other node. The network structure could be described with two incidence matrices in node-link form. The matrix of the incoming links is denoted $A^+$ and the matrix of the outgoing links is denoted $A^-$.

$$A^+_{n,l} = \begin{cases} 1, & l \in \mathcal{I}(n) \text{ (link } l \text{ enters node } n\text{)} \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

$$A^-_{n,l} = \begin{cases} 1, & l \in \mathcal{O}(n) \text{ (link } l \text{ leaves node } n\text{)} \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

**Example:** An example of a simple graph with 4 nodes and 5 links is shown in Figure 1. The numbers in parenthesis stand for the node and link labels. The values associated to the links stand for the communication prices. The matrices $A^-$ and $A^+$ for this graph are:

$$A^- = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad A^+ = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 \end{pmatrix}$$

### 2.2. Multicommodity network flow

We have used a multicommodity network flow model, which is widely used in the literature of network flow routing and optimization [4, 3, 19]. In the multicommodity network flow model, each node can send different pieces of data to any node. Each requested data transfer through the network is called the communication demand $m$ and the set of all communication demands is labelled as $\mathcal{M}$. From the nature of the multicommodity flow model, the data flow of each communication demand can be fragmented into more paths across the network. The model assumes that the data flow is lossless in the network and that it satisfies the flow conservation law at each node.

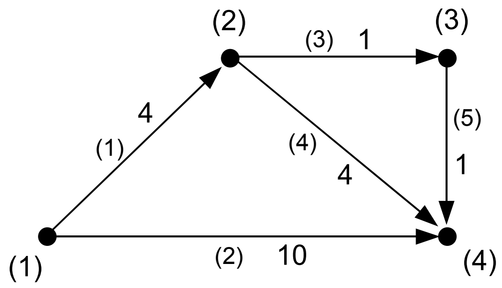


**Figure 1. Graph of the basic network.**

The communication demands can be seen as various flows coming into the network in some nodes and leaving the network in other nodes. Each demand can come into the network in more than one node and leave the network in more nodes too (multi-source, multi-sink). Let us denote the flow amount of demand $m$ coming into the network in node $n$ as $s_{in,n}^{(m)} \geq 0$ and similarly the flow leaving the network in node $n$ as $s_{out,n}^{(m)} \geq 0$. We define the vectors of the data flow of demand $m$ leaving the network as $\bar{s}_{out}^{(m)} \in R^N$ and the data flow incoming into the network $\bar{s}_{in}^{(m)} \in R^N$ over all nodes.

Let $x_l^{(m)} \geq 0$ be the flow of demand $m$ routed through the link $l$. We call $\bar{x}^{(m)} \in R^L$ the flow vector for demand $m$, which describes the flow of the demand in all links over the network. In each node, the flow vector and the leaving/incoming flow have to satisfy the flow conservation law:

$$A^- \bar{x}^{(m)} + \bar{s}_{out}^{(m)} = A^+ \bar{x}^{(m)} + \bar{s}_{in}^{(m)} \qquad \forall m \in \mathcal{M} \quad (3)$$

Finally, we focus on communication capacity constraints. Let $t_l = \sum_{m \in \mathcal{M}} x_l^{(m)}$ be a total amount of data flow in the link $l$ over all communication demands. Then denote vector $\bar{t} \in R^L$ as the total flow for each link over the network. There could be many different capacity constraints in the network (e.g. fixed link capacity, node capacity, etc.). The capacity constrain can by written in matrix form as: $D\bar{t} \leq \bar{\mu}$. Where $\bar{\mu}$ is the limit of the constraints and matrix $D$ represents the constraints structure. When there is a separate capacity for each link, matrix $D$ is the identity matrix of size $[L \times L]$ and $\bar{\mu}$ would just be the capacities of links.

In summary, our network flow model imposes the following group of constraints on the network flow variables $\bar{x}^{(m)}$, $\bar{s}_{in}^{(m)}$, $\bar{s}_{out}^{(m)}$ and $\bar{t}$:

$$\begin{aligned} A^- \bar{x}^{(m)} + \bar{s}_{out}^{(m)} &= A^+ \bar{x}^{(m)} + \bar{s}_{in}^{(m)} \ \forall m \in \mathcal{M} \\ \bar{t} &= \sum_{m \in \mathcal{M}} \bar{x}^{(m)} \\ D\bar{t} &\leq \bar{\mu} \end{aligned} \qquad (4)$$

$$\bar{x}^{(m)} \geq \bar{0}; \quad \bar{s}_{in}^{(m)} \geq \bar{0}; \quad \bar{s}_{out}^{(m)} \geq \bar{0}; \quad \bar{\mu} \geq \bar{0}$$

This model describes the average behavior of the data transmission, i.e., the average data rates on the communication links, and ignores packet-level details of transmission protocols. The link layer communication protocol (e.g. TDMA) should set the bandwidths for each demand according to the flow vectors $\bar{x}^{(m)}$. The link capacity should be defined appropriately, taking into account packet loss and retransmission, so the flow conservation law holds with sufficient probability.

**Example (continued):** Suppose all the link capacities in our example to be equal to 1. Then the capacity constraints matrix $D$ is the identity matrix of size $[5 \times 5]$ and $\bar{\mu} = (1, 1, 1, 1, 1)^T$.

Let there be two communication demands both with flow equal to 1. The first is routed from node 1 to node 4 and the second from node 2 to node 4. Therefore, we have: $\bar{s}_{in}^{(1)} = (1, 0, 0, 0)^T$, $\bar{s}_{in}^{(2)} = (0, 1, 0, 0)^T$, $\bar{s}_{out}^{(1)} = \bar{s}_{out}^{(2)} = (0, 0, 0, 1)^T$.

### 2.3. Routing optimization

All routings that satisfy the system of inequalities (4) are feasible solutions of the routing problem. However, the best solution in terms of some price (e.g. energy consumption, network lifetime) needs to be found. There are plenty of possible price functions, which could be used to determine which solution is the best one. We have focused on convex functions where the cheapest routing in terms of the price can be found in polynomial time. For example, one of the most common price functions used in the sensor network literature is total energy consumption:

$$f_{total\ cost} = \bar{c}^T \bar{t} \qquad (5)$$

Where vector $\bar{c}$ is the vector of the communication prices per data unit for all links in the network. The task of the total energy minimization is to minimize the function $f_{total\ cost}$ by setting the optimal flow vector $\bar{x}$, subject to the system of inequalities (4).

**Example (continued):** Let the communication prices in our example be the same as in Figure 1 (the number associated edges). $\bar{c} = (4, 10, 1, 4, 1)$

One of the optimal solutions for this example is: $\bar{x}^{(1)} = (1, 0, 0, 1, 0)^T$, $\bar{x}^{(2)} = (0, 0, 1, 0, 1)^T$, which means that the first flow is routed through the nodes ($1 \rightarrow 2 \rightarrow 4$) and the second flow is routed through the nodes ($2 \rightarrow 3 \rightarrow 4$). The total flow is the sum of the routings over all demands $\bar{t} = (1, 0, 1, 1, 1)^T$ and the communication price is equal to 10. This energy optimal routing is shown in Figure 2.

## 3. Real-Time multicommodity network flow model

In this section, we have extended the multicommodity flow model by real-time constraints, which guarantee suf-

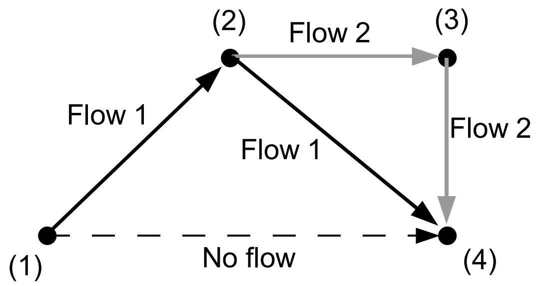


**Figure 2. An example of optimal data flow routing with capacity constraints.**

ficient routing delay through the network. Each communication demand has its own deadline and the communication delay of this demand has to be shorter than the deadline. The hop delay is measured from the moment when the processor in one routing device starts to send the message until the moment when the processor in the neighboring routing device receives the message, i.e. this delay includes the transmission delay, MAC delay, and the delays in routing devices. We have modeled the hop delay as an integer value associated to each communication link. For transparent model derivation, we assume the same communication delay over the entire network (i.e. each communication hop causes delay equal to one). However, the model could be easily extended to the general form (see [16] for more details), where the communication delays can be different integer values for each link and demand.

**Example (continued):** We added the deadline constraints into our example. Namely, the first communication demand has the deadline equal to 2 communication hops and the second communication demand has the deadline equal to 1 communication hop. With these new constraints, the solution shown in Figure 2 is not feasible.

### 3.1. Mathematical model of real-time routing

Let vector $\bar{x}^{(m,k)} \in R^L$ denote the flow of communication demand $m$ with integer communication delay $k$ in the network. Then the flow vector ($\bar{x}^{(m)}$) independent to the flow delay of demand $m$ is equal to the sum of the flow vectors over all acceptable delays: $\bar{x}^{(m)} = \sum_{k=0}^{d^{(m)}} \bar{x}^{(m,k)}$. Where $d^{(m)}$ denotes the deadline of the communication demand $m$. Using this equation, we can rewrite the equation for total flow vector from the system of inequalities (4) into a new form:

$$\bar{t} = \sum_{m \in \mathcal{M}} \sum_{k=0}^{d^{(m)}} \bar{x}^{(m,k)} \qquad (6)$$

Vector $\bar{s}_{out}^{(m,k)} \in R^N$ stands for the flow of the demand $m$ leaving the network with communication delay $k$ and vector $\bar{s}_{in}^{(m,k)} \in R^N$ denotes the flow of demand $m$ coming into the network with initial delay $k$. As usual, the flow of each demand may come into the network and leave it in more nodes. If all flows of one demand coming into the network in different nodes have the same initial delay, then $\bar{s}_{in}^{(m,0)} = \bar{s}_{in}^{(m)}$, and $\bar{s}_{in}^{(m,k)} = 0$ for $k > 0$. The flow of demand $m$ leaving the network prior the deadline is:

$$\bar{s}_{out}^{(m)} = \sum_{k=0}^{d^{(m)}} \bar{s}_{out}^{(m,k)} \qquad \forall m \in \mathcal{M} \qquad (7)$$

Through the Equations (6, 7), we have converted the real-time constraint (i.e. the delay has to be shorter than the deadline) to the structural constraint. Only the flows, where their delays are shorter than their deadlines, are represented. The flows, which do not meet the deadline, cause that the flow conservation law does not hold and then the network flow constraints are not satisfied, i.e. this solution is not feasible.

If the flow is sent through the network, the flow delay is increased by each communication hop. The flow of demand $m$ coming into node $n$ with communication delay $k$ has to either leave the network in node $n$ with the same delay $k$ or reach the neighbor node with delay $k+1$. The flow conservation law from Equation (3) can be rewritten in the delay awareness form as:

$$\begin{gathered} A^{-}\bar{x}^{(m,k+1)} + \bar{s}_{out}^{(m,k)} = A^{+}\bar{x}^{(m,k)} + \bar{s}_{in}^{(m,k)} \\ \forall m \in \mathcal{M},\ 0 \le k \le d^{(m)} \end{gathered} \qquad (8)$$

In summary, the constraints of the real-time multicommodity flow routing problem can be written as:

$$\begin{aligned} & A^{-}\bar{x}^{(m,k+1)} + \bar{s}_{out}^{(m,k)} = A^{+}\bar{x}^{(m,k)} + \bar{s}_{in}^{(m,k)} \\ & \qquad \forall m \in \mathcal{M},\ 0 \le k \le d^{(m)} \\ & \bar{s}_{out}^{(m)} = \sum_{k=0}^{d^{(m)}} \bar{s}_{out}^{(m,k)} \qquad \forall m \in \mathcal{M} \\ & \bar{t} = \sum_{m \in \mathcal{M}} \sum_{k=0}^{d^{(m)}} \bar{x}^{(m,k)} \\ & D\bar{t} \le \bar{\mu} \\ & \bar{x}^{(m,k)} \ge \bar{0}; \quad \bar{x}^{(m,0)} = \bar{0}; \quad \bar{s}_{in}^{(m,k)} \ge \bar{0}; \\ & \bar{s}_{out}^{(m,k)} \ge \bar{0}; \quad \bar{\mu} \ge \bar{0} \end{aligned} \qquad (9)$$

All feasible routings, which obey the deadlines and capacity constraints and realize all communication demands are described by the system of inequalities (9). To choose the cheapest one in terms of the price function (e.g. 5) we can use Linear Programming with these constraints.

**Example (continued):** Solving the example with the deadline constraints according to (9) and price function (5) we get the new solution: $\bar{x}^{(1,1)} = (0,1,0,0,0)^T$, $\bar{x}^{(1,2)} = (0,0,0,0,0)^T$, $\bar{x}^{(2,1)} = (0,0,0,1,0)^T$, $\bar{x}^{(1,0)} = \bar{x}^{(2,0)} = (0,0,0,0,0)^T$. This means that the first flow is routed through nodes ($1 \rightarrow 4$) and the second flow through nodes ($2 \rightarrow 4$).

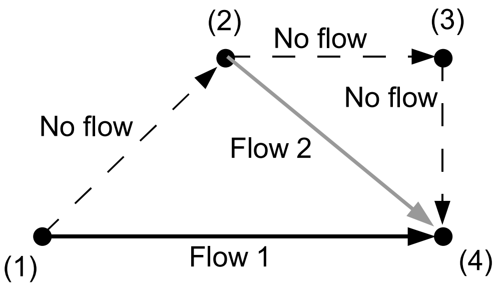


**Figure 3. An example of optimal data flow routing with capacity and deadline constraints.**

$\bar{s}_{out}^{(1,1)} = (0,0,0,1)^T$ and $\bar{s}_{out}^{(1,0)} = \bar{s}_{out}^{(1,2)} = (0,0,0,0)^T$, which means that the first flow leaves the network in node 4 with communication delay 1 and no part of the first flow leaves the network with communication delays 0 and 2. Similarly for the second flow $\bar{s}_{out}^{(2,1)} = (0,0,0,1)^T$ and $\bar{s}_{out}^{(2,0)} = (0,0,0,0)^T$. $\bar{s}_{out}^{(2,2)}$ has no sense for the second flow because its deadline is equal to 1.

The total load of the links is: $\bar{t} = (0,1,0,1,0)^T$ and the communication price is equal to 14. The energy optimal real-time routing is shown in Figure 3.

### 3.2. Matrix form of real-time routing

For a more transparent description we have presented a matrix formulation of the inequalities (9) in this section.

We defined a new column vector for the network flow of demand $m$ from vectors $\bar{x}^{(m,k)}\ \forall k = 1 \dots d^{(m)}$ as:

$$\bar{y}^{(m)} = (\bar{x}^{(m,1)^T}, \bar{x}^{(m,2)^T} \dots \bar{x}^{(m,d^{(m)})^T})^T \qquad (10)$$

and new column vector for routing demands for each demand $m$ from vectors $\bar{s}^{(m,k)}$ as:

$$\begin{aligned} \bar{z}^{(m)} &= (\bar{s}_{out}^{(m,0)^T} \dots \bar{s}_{out}^{(m,d^{(m)})^T})^T - \\ &- (\bar{s}_{in}^{(m,0)^T} \dots \bar{s}_{in}^{(m,d^{(m)})^T})^T \end{aligned} \qquad (11)$$

To rewrite the flow conservation law (8), we defined matrix $A^{(m)}$ for each demand $m$. The structure of the matrix $A^{(m)}$ is the same for all demands $m$, however the size of the matrix depends on the deadline $d^{(m)}$ of the demand $m$. For demands with the same deadlines, the matrix $A^{(m)}$ are identical.

$$A^{(m)} = \begin{pmatrix} -A^- & 0 & . & 0 & 0 \\ A^+ & -A^- & . & 0 & 0 \\ 0 & A^+ & . & 0 & 0 \\ . & . & . & . & . \\ 0 & 0 & . & A^+ & -A^- \\ 0 & 0 & . & 0 & A^+ \end{pmatrix} \qquad (12)$$

The size of the matrix $A^{(m)}$ depends on deadline $d^{(m)}$ of the demand $m$ and is $\left[N \cdot \left(d^{(m)} + 1\right) \times L \cdot d^{(m)}\right]$ where $[N \times L]$ is the size of matrices $A^+$ and $A^-$.

The total link load (6) can be rewritten for the network flow variable $\bar{y}^{(m)}$ in the new form as: $\bar{t} = \sum_{m \in \mathcal{M}} G^{(m)} \bar{y}^{(m)}$. Where matrix $G^{(m)}$ consists of the identity matrices $I$ as $G^{(m)} = (I\ I\ ...\ I)$ and its size is $\left[L \times L \cdot \left(d^{(m)} - 1\right)\right]$.

The problems of the real-time multicommodity flow routing minimizing the energy consumption (constrained by the system of inequalities (9)), can be written in the matrix form as a linear optimization problem:

$$\min \quad \bar{c}^T \bar{t}$$

**subject to:**

$$\begin{array}{rcll} A^{(m)} \bar{y}^{(m)} & = & \bar{z}^{(m)} & \forall m \in \mathcal{M} \\ B^{(m)} \bar{z}^{(m)} & = & \bar{s}_{out}^{(m)} - \bar{s}_{in}^{(m)} & \forall m \in \mathcal{M} \\ D\bar{t} & \leq & \bar{\mu} & \\ \bar{t} & = & \sum\limits_{m \in \mathcal{M}} G^{(m)} \bar{y}^{(m)} & \\ \bar{y}^{(m)} \geq \bar{0}; & \bar{s}_{in}^{(m)} \geq \bar{0}; & \bar{s}_{out}^{(m)} \geq \bar{0}; & \bar{\mu} \geq \bar{0} \end{array} \qquad (13)$$

Where the matrix $B^{(m)}$ consists of the identity matrix $I$ and $B^{(m)} = (I\ I\ ...\ I)$ with the size $\left[N \times N \cdot d^{(m)}\right]$. Then the vector $\bar{y}^{(m)}$ describes the optimal real-time routing in the link-delay space for the communication demand $m$ and vector $\bar{z}^{(m)}$ describes the deliveries of the flow of the demand $m$ to its destinations. This problem can be solved by the linear programming method. If there is no way how to route the communication demands with the deadlines through the network, the problem (13) has no feasible solution.

**Example (continued):** The matrix $A^{(m)}$ for the first demand with the deadline equal to 2 is:

$$A^{(1)} = \begin{pmatrix} -A^- & 0 \\ A^+ & -A^- \\ 0 & A^+ \end{pmatrix}$$

and this matrix corresponds to the expanded graph in Figure 4.

If we solve our example according to the system of inequalities (13), the optimal solution is: $\bar{y}^{(1)} = (0,1,0,0,0,0,0,0,0,0)^T$, $\bar{y}^{(2)} = (0,0,0,1,0)^T$ and $\bar{z}^{(1)} = (0,0,0,0,0,0,0,1,0,0,0,0)^T$, $\bar{z}^{(2)} = (0,0,0,0,0,0,0,1)^T$, which is only a different record of the solution shown in Figure 3 according to the system of inequalities (9). (the transformation is described by Equations (10) and (11)).

### 3.3. Intuitive presentation of extended graph

In this section, we have illustrated in an intuitive way the graph transformation, which has been discussed in the previous sections by mathematical equations. New variables have appeared for each communication link as well

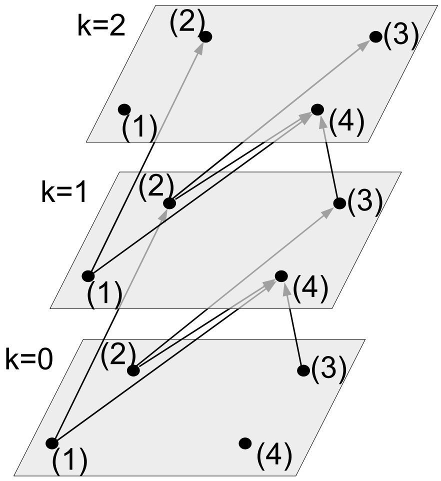


**Figure 4. Expanded graph for 2 hops.**

as new constraining equations for each node. These variables and constraints can be seen as virtual layers of the network where each layer represents a different communication delay $k$. The number of the network layers is equal to the integer deadline of the demand $m$ plus one (the number of allowed communication hops plus a zero layer). As consistent with the structure of matrix $A^{(m)}$ (12), all communication links are redirected to the nodes in the higher layer, which means that the flow is routed not only in node space but also in delay space. Because the number of layers is limited by the deadline and the flow can leave the network only in virtual nodes of the destination nodes, all possible routings through this transformed network hold the deadlines. An example of the expanded graph from Figure 1 is shown in Figure 4.

### 3.4. Simultaneous real-time and non-real-time routing

The network communication problems often comprise real-time and non-real-time communication demands. The non real-time routing demands can be taken as if they would have no communication delays. Therefore, they can be solved as the common multicommodity routing problem presented in Section 2. The only resources where the non real-time and the real-time flows interact are guarded by the capacity constraints. Let $\mathcal{P}$ be a set of non real-time demands and vector $x^{(p)}$ be a non real-time flow of demand $p$.

A new equation of flow conservation for non real-time flow has to be added to the system of inequalities (13):

$$(A^{+} - A^{-})\bar{x}^{(p)} = \bar{s}^{(p)}_{out} - \bar{s}^{(p)}_{in} \qquad p \in \mathcal{P} \qquad (14)$$

and the equation for the total link load has to be changed to consider the non real-time flow:

$$\bar{t} = \sum_{m \in \mathcal{M}} G^{(m)} \bar{y}^{(m)} + \sum_{p \in \mathcal{P}} \bar{x}^{(p)} \qquad (15)$$

With these changes of the system of inequalities (13), simultaneous real-time and non real-time routing is possible.

### 3.5. Computation complexity

A big advantage of this approach is the polynomial computation complexity. The exact time complexity of the computation depends on a specific solver used for the linear optimization and on the number of variables and constraints. We denote $N$ as the number of nodes, $L$ as the number of communication links, $K_{max}$ as the maximum of deadlines $d^{(m)}$ and $M$ as the number of communication demands. The number of variables $no_{var}$ is the sum of the flow variables $y$, $z$ and $t$:

$$no_{var} = L(\sum_{m=1}^{M}(d^{(m)})) + N \sum_{m=1}^{M}(d^{(m)} + 1) + L \quad (16)$$

If we consider that $L \leq N(N-1)$ and that the number of allowed communication hops is smaller than the number of nodes ($K_{max} \leq N - 1$), we can write the order of the worst case of the variables complexity as:

$$O_{var}(MN^3) \qquad (17)$$

### 3.6. Integral flows

In the case that the data flow cannot be fragmented (e.g. the data are transferred in packet, which cannot be fragmented), the multicommodity network flow problem is NP-hard even for two communication demands (see [8]). The multicommodity routing problem with integral flows can be solved through algorithms based on the Branch-and-Bound mechanism, Cutting planes, Lagrangian relaxation, Evolution algorithms, etc., where the continuous problem presented in this article is used as a heuristic (see [3, 15]). The second possibility, adopted in our experiments, is the usage of solvers for integer linear programming (e.g. CPLEX). For large networks with integral flow, some approximation algorithm for multicommodity flow can be used. (see e.g. [17])

## 4. Numerical experiments

To demonstrate the benefits and correctness of our approach for real-time routing, we simulated the routing problems in Matlab. We have demonstrated the data collection problem in the network with 100 nodes for two data flows and a set of experiments to demonstrate the time complexity of the computation algorithm.

### 4.1. Data collection problem

For the data collection problem we consider a network field of size $[10 \times 10]$ and divide it into 100 subsquares of size $[1 \times 1]$. One node is randomly placed into each subsquare and the communication distance is set to 2. The communication links are set between the nodes within the communication distance which ensure that each node inside the field has at least 3 communication links to its neighbors. An example of such a random network topology is shown in Figure 5. Each node has a 50% probability that it will send data of the first type to the first destination node and a 50% probability that it will send data of

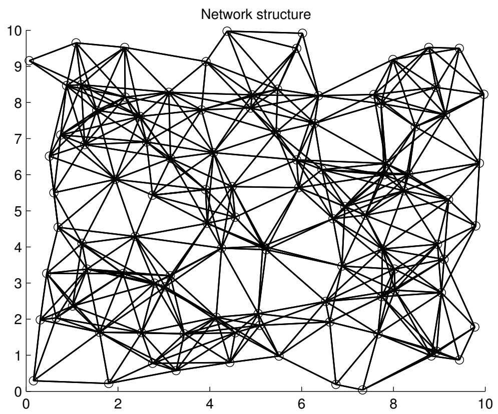


**Figure 5. Topology of randomly generated wireless network with 100 nodes and 962 directed communication links.**

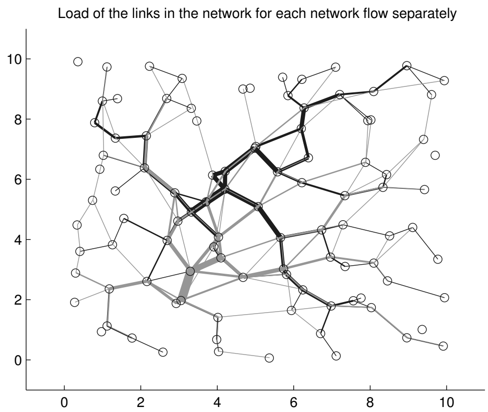


**Figure 6. Optimal real-time data flow routing in a network with 100 nodes and two data flows for the data collection problem.**

the second type to the second destination node. Each data flow is equal to 1 (i.e. there are nodes which send no data flow, nodes which send flow equal to 1 and nodes which send flow equal to 2). The deadline of the first flow is set to 8 communication hops and the deadline of the second flow is set to 6 communication hops. The link capacities are set to 30. The communication prices per transmitted data flow unit are set as the power of the distance between the nodes.

The resulting data flow routing through the network is shown in Figure 6, where only the used links are shown and the links width is a logarithmic function of the amount of the data flow routed through the link. The first type of data flow is in black and the second data flow is in grey. Each routing path of the first type of data flow has at maximum 8 communication hops and each routing path of the second type of data flow has at maximum 6 communication hops.

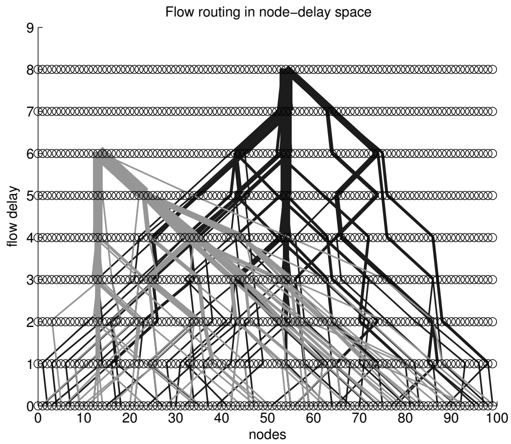


**Figure 7. Optimal real-time data flow routing in node-delay space for two communication demands with deadlines 8 and 6 communication hops for the data collection problem. The flow destination nodes are nodes number 54 and 13.**

The routing in the node-delay space is shown in Figure 7. On the vertical axis, the single nodes are placed and on the horizontal axis, the integer delays are placed. The flows, which are produced by the nodes, start with delay equal to zero and through each communication hop the delay of the part of the transmitted flow is increased. The lines width is a logarithmic function of the amount of the flow. The vertical lines in node 54 and in node 13 represent the flow, which is buffered in the destination nodes.

### 4.2. Computation complexity

To demonstrate the computation complexity of our approach we varied the last experiment of data collection. The size of the square field is gradually set from $[2 \times 2]$ to $[10 \times 10]$ so too the number of nodes from 4 to 100. The communication distance is kept at 2 and the communication prices are kept as a power of the nodes distance. The number of communication demands was set to 5 (and to 20 for the second experiment). Each node sends data flow to 5 different destination nodes (or to 20 in the second experiment). Each data flow is equal to 1. The link capacities are set as $\mu = 0.25NM/3$ where $N$ denotes the number of the nodes in the network and $M$ denotes the number of communication demands. The flow deadline for each communication demand is set as $d^{(m)} = \sqrt{N}$.

The simulation has been run 20 times for each number of nodes on randomly constructed networks for both continuous and integral flow. The resulting times are taken as the average values from all feasible instances (see Table 1). The simulation has been run using a commercial solver CPLEX 9.1 on a computer with an AMD processor Opteron 248 at 2200 MHz, 2GB RAM DDR at 400MHz.

**Table 1. Average computation times [s]**

| | Fragmented flow | | Integral flow | |
|---|---|---|---|---|
| Nodes | 5 flows | 20 flows | 5 flows | 20 flows |
| 4 | 0.02 | 0.02 | 0.02 | 0.02 |
| 9 | 0.02 | 0.03 | 0.02 | 0.05 |
| 16 | 0.03 | 0.11 | 0.04 | 0.13 |
| 25 | 0.06 | 0.28 | 0.08 | 0.36 |
| 36 | 0.13 | 0.57 | 0.16 | 0.77 |
| 49 | 0.22 | 1.21 | 0.28 | 1.52 |
| 64 | 0.36 | 2.30 | 0.46 | 2.77 |
| 81 | 0.75 | 3.99 | 0.87 | 5.20 |
| 100 | 1.96 | 7.45 | 1.96 | 8.18 |

## 5. Conclusion

We have focused on real-time routing in multi-hop networks, like sensor networks. We have used the multi-commodity network flow routing problem and extended it to solve real-time routing in general multi-hop network. The work concentrates on a mathematical derivation of the computation algorithm and it sets a task to show whether this approach is usable for solving real-time routing problems. We have derived an off-line computational algorithm and in the experiment section we have shown that it is applicable even for large networks.

There are many possible extensions of our work. One such extension could be to use a more detailed model of the energy consumption per transmission (e.g. [19]) and try to minimize the total energy subject to real-time routing constraints. Another extension could be to pose and solve joint real-time routing and transmission scheduling problems, giving not only the optimal routes but also the optimal TDMA schedule (e.g. optimal allocation of time-slots to links). The approach presented in this article can be applied to many other problems, like routing in wired networks or traffic control. Centralized solutions to these problems are useful for gaining insight in the performance limits of real-time routing, e.g. during system design, but could also give intuition about distributed protocols. At the present time, we are working on the distributed algorithm, which is based on the dual decomposition of the approach presented in this article.

**Acknowledgement:** This work was supported by the Ministry of Education of the Czech Republic under Research Programme MSM6840770038 and by the European Commission under project FRESCOR IST 034026.

## References

[1] T. F. Abdelzaher, S. Prabh, and R. Kiran. On real-time capacity limits of multihop wireless sensor networks. *25th IEEE International Real-Time Systems Symposium (RTSS'04)*, pages 359–370, 2004.

[2] K. Akkaya and M. Younis. Energy and QoS aware routing in wireless sensor networks. *Cluster Computing*, 7(2-3):179–188, July 2005.

[3] D. P. Bertsekas. *Network Optimization: Continuous and Discrete Models*. Athena Scientific, 1998.

[4] D. P. Bertsekas and R. Gallager. *Data Networks*. Prentice Hall, 2004.

[5] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[6] M. Caccamo and L. Zhang. The capacity of implicit EDF in wireless sensor networks. *Real-Time Systems, 2003. Proceedings. 15th Euromicro Conference on*, pages 267–275, July 2003.

[7] O. Chipara, Z. He, G. Xing, Q. Chen, X. Wang, C. Lu, J. A. Stankovic, and T. F. Abdelzaher. Real-time power aware routing in wireless sensor networks. *the 14th IEEE International Workshop on Quality of Service (IWQoS 2006)*, June 2006.

[8] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* Freeman , Oxford, UK, 1979.

[9] T. He, J. Stankovic, C. Lu, and T. Abdelzaher. SPEED: a stateless protocol for real-time communication in sensor networks. *International Conference on Distributed Computing Systems (ICDCS 2003)*, pages 46– 55, May 2003.

[10] M. Johansson and L. Xiao. Scheduling, routing and power allocation for fairness in wireless networks. *IEEE Vehicular Technology Conference*, 3:1355–1360, May 2004.

[11] A. Koubaa, M. Alves, and E. Tovar. GTS allocation analysis in IEEE 802.15.4 for real-time wireless sensor networks. *The 14th International Workshop on Parallel and Distributed Real-Time Systems (WPDRTS'06), Rhodes, Greece*, April 2006.

[12] A. Koubaa, M. Alves, and E. Tovar. i-GAME: An implicit GTS allocation mechanism in IEEE 802.15.4. *The 18th Euromicro Conference on Real-Time Systems (ECRTS'06), Dresden, Germany*, Jul 2006.

[13] R. Mangharam, A. Rowe, and R. Rajkumar. FierFly: a cross-layer platform for wireless sensor networks. *Real-Time Systems Journal, Special Issue on Real-Time Wireless Sensor Networks*, November 2006.

[14] A. Ouorou, P. Mahey, and P. Vial. A survey of algorithms for convex multicommodity flow problems. *Management Science*, 46(1):126–147, January 2000.

[15] M. Pióro and D. Medhi. *Routing, Flow, and Capacity Design in Communication and Computer Networks*. Morgan Kaufmann, 2004.

[16] J. Trdlička. Real-time network flow routing. Technical Report 0.2, Department of Control Engineering, Faculty of Electrical Engineering, CTU Prague, Czech Republic, 2007. http://rtime.felk.cvut.cz/wiki/images/4/46/Real-time_routing_0.2.pdf.

[17] V. V. Vazirani. *Approximation Algorithms*. Springer-Verlag, Berlin, Germany, 2003.

[18] T. Watteyne and I. Auge-Blum. Proposition of a hard real-time MAC protocol for wireless sensor networks. *Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, 2005. 13th IEEE International Symposium on*, pages 533– 536, September 2005.

[19] L. Xiao, M. Johansson, and S. Boyd. Simultaneous routing and resource allocation via dual decomposition. *IEEE Transactions on Communications*, 52(7):1136–1144, July 2004.