Paper

# Fairness considerations with algorithms for elastic traffic routing

Tibor Cinkler, Péter Laborczi, and Michał Pióro

**Abstract—The bit rate of modern applications typically varies in time. We consider the traffic elastic if the rate of the sources can be controlled as a function of free resources along the route of that traffic. The objective is to route the demands optimally in sense of increasing the total network throughput while setting the rates of sources in a fair way. We propose a new fairness definition the relative fairness that handles lower and upper bounds on the traffic rate of each source and we compare it with two other known fairness definitions, namely, the max-min and the proportional rate fairness. We propose and compare different routing algorithms, all with three types of fairness definitions. The algorithms are all a tradeoff between network throughput, fairness and computational time.**

***Keywords—elastic traffic, routing, fairness, maximum throughput, algorithms, ILP, heuristics.***

## 1. Introduction

In modern infocommunications networks the rate of sources typically varies in time. On the one hand this is due to silence period detection of voice codecs, compression of voice and video to variable bit rate depending on the amount of information to be carried. On the other hand, the bit rate of the data that is not sensitive to delay and delay variation can be tuned according to the network conditions to maximise the throughput without affecting the delay sensitive traffic.

In the new Internet architecture there is a growing interest in devising bandwidth sharing algorithms, which can cope with a high bandwidth utilisation and at the same time maintain some notion of *fairness*, such as the max-min (MMF) [1, 2] or proportional rate fairness (PRF) [3].

Examples of elastic traffic are TCP sessions in IP networks and available bit rate (ABR) service class in asynchronous transfer mode (ATM) networks. Label switch paths (LSPs) of multiprotocol label switching (MPLS) networks are also easy to reconfigure. In all cases the rate of sources is influenced by the load of the network.

Three variants of elastic traffic optimisation can be distinguished: (1) *fixed paths*, (2) *pre-defined paths* or (3) *free paths* can be assumed. In the *fixed paths* case there is a single path defined between each origin-destination (O-D) pair and the allocation task is to determine the bandwidth assigned to each demand. In the *pre-defined paths* case we assume that between each O-D pair there is a set of admissible paths, that can be potentially used to realize the flow of the appropriate demand. In this case the allocation task does not only imply the determination of the bandwidth of the flow, but also the identification of the specific path that is used to realize the demands [4]. In the *free paths* case there is no limitation on the paths, i.e., the task is to determine the bandwidth of the traffic AND the routes used by these demands simultaneously. This novel approach, the joint path and bandwidth allocation is the main topic of this article.

Recent research results indicate that it is meaningful to associate a minimum and maximum bandwidth even with elastic traffic [5], therefore it is important to develop models and algorithms for this type of services. As an example the ABR service can be mentioned that has the minimum cell rate (MCR) lower bound and the peak cell rate (PCR) upper bound. For the bounded elastic services we propose a special weighted case of MMF notion: relative fairness (RF) that maximises the minimum rates relative to the difference between upper and lower bounds for each demand.

Considering literature, different aspects of the max-min fairness policy have been discussed in a number of papers, mostly in ATM ABR context, since the ATM Forum adopted the max-min fairness criterion to allocate network bandwidth for ABR connections, see, e.g., [6, 7]. However, these papers do not consider the issue of path optimisation in the bounded elastic environment. MMF routing is the topic of the paper [8], where the widest-shortest, shortest-widest and the shortest-dist algorithms are studied. These algorithms do not optimise the path allocation. A number of fairness notions are discussed and associated optimisation tasks are presented in [5] for the case of unbounded flows and assuming fixed routes.

Proportional rate fairness is proposed by Kelly [3] and also summarised by Massoulie and Roberts in [5]. The objective of PRF is to maximise the sum of logarithms of traffic bandwidths. While [3] does consider the path optimisation problem, it does not focus on developing an efficient algorithm for path optimisation when the flows are bounded.

Recent research activities focused on allocating the bandwidth of fixed paths. In [4] the approach has been extended such that not only the bandwidth, but also the paths are chosen from a set of pre-defined paths. The formulation of the pre-defined path optimisation problem is advantageous, since it has significantly less variables than the free path optimisation. However, its limitation is that the whole method relays on the set of pre-defined alternative paths. If the set of paths are given in advance, setting up elastic source rates in fair way leads to suboptimal solution. Better results can be achieved if we determine the rate of elastic sources AND the routes used by these demands simultaneously. There arises a question how much resources should be reserved for each demand, and what path should be chosen for carrying that traffic in manner to utilise resources efficiently while obeying fairness constraints as well. In this paper we investigate these questions and propose exact algorithms for solving it, assuming three types of fairness definition: RF, MMF and PRF.

- *Relative fairness.* In this case the aim is to increase the rates relative to the difference between upper and lower bounds for each demand. RF is a useful subcase of the bounded MMF definition, which can be solved in shorter running time.
- *Max-min fairness.* In this case we want to maximise the smallest demand bandwidth.
- *Proportional rate fairness.* In this notion the aim is to set the rates as a result of a convex optimisation, prioritising shorter paths to longer ones.

For each of these fairness definitions, a parameter ($\alpha$, $\beta$ and $\gamma$, respectively) is associated with each source expressing the bandwidth of the source. In case of RF parameter $\alpha_d$ indicates the rate of source $d$ relative to the difference between upper and lower bounds. In case of MMF parameter $\beta_d$ indicates the bandwidth of demand $d$. Parameters $\alpha_d$ and $\beta_d$ can be unique for all sources (denoted by $\alpha$ and $\beta$, called *uniform parameter* case), which is optimal in sense of fairness, however, typically the parameter of some demands can be increased (called *different parameters* case), which increases the rate of some sources while it does not limit the rate of other sources. The value $\min_d(\alpha_d)$ is simply denoted by $\min(\alpha)$ and $\sum_d(\alpha_d)/D$ is denoted by $\mathrm{av}(\alpha)$ where $D$ is the number of demands in the network. Analogous notation is used for $\beta$. In case of PRF parameter $\gamma$ is unique for the whole network: it indicates the sum of logarithms of traffic bandwidths.

All these fairness definitions can be investigated in the bounded case (bounds on the minimal and maximal bandwidths for each O-D pairs). In the unbounded case MMF and PRF can be optimised, while RF has no sense without bounds. All fairness definitions can be formalised with unweighted and weighted fairness measures. We formulate the unweighted case, i.e., assume that all sources have the same priority, and then extend the model for the weighted case, i.e., when the sources have different priorities.

Accordingly, the following cases will be considered in the following sections:

– relative fairness with bounds with uniform parameter (RF/B/U): in Section 2;

– relative fairness with bounds with different parameters (RF/B/D): in Section 2.2;

– max-min fairness without bounds with uniform parameter (MMF/NB/U): in Section 3;

– max-min fairness without bounds with different parameters (MMF/NB/F): in Section 3.2;

– max-min fairness with bounds with uniform parameter (MMF/B/U): in Section 3.4;

– max-min fairness with bounds with different parameters (MMF/B/F): in Section 3.4;

– proportional rate fairness without bounds (PRF/NB): in Section 4;

– proportional rate fairness with bounds (PRF/B): in Section 4.

First, we focus on the basic case of relative fairness with bounds and uniform parameter (RF/B/U) and we further enhance the method to increase network throughput by utilising the spare resources (RF/B/D). The exact formulation of the problem is presented and methods are proposed which solve them to required accuracy.

## 2. Relative fairness: formulation and algorithms

In this section relative fairness is considered that maximises the minimum rates relative to the difference between upper and lower bounds for each demand. The formulation relays on the integer linear programming (ILP) formulation of the unsplittable minimal cost multicommodity flow problem.

The network topology of $N$ nodes and $L$ links with link capacities $C_l$ $(l = 1,2,...,L)$ are given. The lower and the upper bounds for demands $d = 1,2,...,D$ are respectively $m_d$ and $M_d$. Output is the capacity requirement (bandwidth) $b_d$ of demand $d$: $m_d \leq b_d \leq M_d$, where $b_d$ can be expressed as $b_d = m_d + \alpha(M_d - m_d)$ and where $\alpha$ (the parameter of RF) is a continuous variable which ensures fairness. It can take values $0 \leq \alpha \leq 1$. In this formulation we assume that it has the same value for all demands $d = 1,2,...,D$. A 0-1 flow indicator variable on link $l$ of demand $d$ is $x_l^d$.

Objective:

$$\max \quad \alpha\,. \tag{1}$$

Subject to constraints:

$$\sum_d x_l^d \cdot (m_d + \alpha(M_d - m_d)) \leq C_l \quad l = 1,2,..,L, \tag{2}$$

where

$$0 \leq \alpha \leq 1\,, \tag{3}$$

$$\sum_{j=1}^{N} x_{ij}^d - \sum_{k=1}^{N} x_{ki}^d = \begin{cases} 1 & \text{if } i \text{ is the source of } d \\ -1 & \text{if } i \text{ is the sink of } d \\ 0 & \text{otherwise} \end{cases}, \quad i = 1,2,...,N, d = 1,2,...,D \tag{4}$$

$$x_l^d \in \{0,1\}, l = 1,2,...,L, d = 1,2,...,D\,. \tag{5}$$

Equations (2) are capacity constraints and Eqs. (4) are the well known flow-conservation constraints. Unfortunately, this is a *nonlinear* formulation, since constraint (2) is not linear. In the following subsections it will be linearised by a simple method.

### *2.1. Algorithms for a single $\alpha$ for the whole network (RF/B/U)*

For configuring networks which handle elastic traffic heuristic methods are preferred since nonlinearity is hard to handle. However, in this case the following simple deterministic algorithm guarantees the quality of the results.

#### 2.1.1. Binary search algorithm (BSA)

This algorithm is based on the idea of binary search for finding the optimal value of $\alpha$ between 0 and 1.

*Step 1*: Check the feasibility by setting $\alpha = 0$. If satisfied, check the upper bounds by setting $\alpha = 1$. If satisfied, the solution is obtained, if not, set iteration counter $k = 1$, $\alpha = 0$, $\Delta = 1/2$ and proceed to Step 2.

*Step 2*: Set $\alpha = \alpha + \Delta$ and run the unsplittable multicommodity flow (UMCF) subroutine (see Section. 2.1.2).

*Step 3*: Increment $k$. If UMCF was feasible set $\Delta = 1/2^k$ else set $\Delta = -1/2^k$.

*Step 4*: Go to Step 2 until required fairness is achieved.

This deterministic method guarantees the quality of the results, i.e., if the number of iterations is $k$, then the largest "unfairness" in sense of parameter $\alpha$ is upper bounded by $1/2^k$. In the $7^{th}$ iteration this unfairness will be less than 1% (0.0078125), while in the $10^{th}$ iteration less than $10^{-3}$.

#### 2.1.2. The unsplittable multicommodity flow subroutine

This subroutine finds the optimal routing for fixed $\alpha$. This is the unsplittable multicommodity flow problem referred to as UMCF. It can be solved by an ILP solver, e.g., CPLEX. Set:

$$b_d = (m_d + \alpha(M_d - m_d)) \quad d = 1,2,...,D. \tag{6}$$

Objective:

$$\min \sum_d b_d \sum_l x_l^d . \tag{7}$$

Subject to constraints (4), (5) and:

$$\sum_d b_d x_l^d \le C_l \quad l = 1,2,...,L. \tag{8}$$

#### 2.1.3. Adaptive search algorithm (ASA)

Instead of the BSA a faster method can be used for setting value of $\alpha$. This is an extension of BSA referred to as ASA. The idea is to increase $\alpha$ without changing the paths. After a feasible UMCF subroutine we find a new value of $\alpha^{(k+1)}$ to be used in the forthcoming $(k+1)^{th}$ iteration, based on the paths of the current $k^{th}$ iteration. The new alpha is calculated by the following equation derived from constraint (2):

$$\alpha^{(k+1)} = \min_l \left\{ \frac{C_l - \sum_d m_d x_l^{d,(k)}}{\sum_d x_l^{d,(k)} (M_d - m_d)} \right\} \; l = 1,2,...,L. \tag{9}$$

This increase of parameter $\alpha$ is carried out after each feasible UMCF subroutine. Adaptive search speeds up the algorithm or increases the precision of $\alpha$.

### *2.2. Allowing slightly different values of $\alpha$ within a network (RF/B/D)*

Since all traffics are changed equally according to the definition of parameter $\alpha$, the first saturated link will limit the value of $\alpha$. Therefore, an iterative approach is needed, which increases the network throughput, however, it deteriorates the fairness slightly, by offering more resources to demands not using saturated links. The idea is to set a new, higher value of $\alpha^{(k)}$ $(k = 1,2,...)$ for some demands by using free resources of yet unsaturated links in each iteration $k$. Note, that there are two alternatives:

*Case 1*: The paths of demands are determined in the first iteration. They are not changed any more, only the bandwidths.

*Case 2*: Both, the paths and bandwidths are improved in each iteration.

#### 2.2.1. Case 1: Increase bandwidth

In this case the paths assigned to demands are determined within the first phase and are not changed any more. The allocations are changed only according to the following algorithm ($Y_l^k$ represents the free capacity on link $l$ after the $k^{th}$ iteration):

*Step 1*: Set $k = 0$, $\alpha^{(0)} = \alpha$, $b_d = m_d + \alpha^{(k)}(M_d - m_d)$, $Y_l^{(0)} = C_l - \sum_d b_d x_l^d$.

*Step 2*: Set $k$++.

*Step 3*: Remove all saturated links and paths using these links.

*Step 4*: If there is no more demand left or $\alpha^{(k-1)} = \alpha^{(k-2)}$ then *Stop*, otherwise continue.

*Step 5*:

$$\alpha^{(k)} = \min_l \left\{ \frac{Y_l^{(k-1)} - \sum_d m_d x_l^d}{\sum_d x_l^d (M_d - m_d)} \right\} . \tag{10}$$

*Step 6*:

$$b_d = m_d + \alpha^{(k)}(M_d - m_d)\,. \tag{11}$$

*Step 7*:

$$Y_l^{(k)} = Y_l^{(k-1)} - \sum_d b_d x_l^d\,.$$

*Step 8*: Go to Step 2.

The new value for $\alpha$ is calculated by Eq. (10) that has analogous meaning to Eq. (9). Note, that this iterative procedure has to be repeated up to $L$ times, where $L$ is the number of links in total for the considered network, since each iteration will saturate at least one link.

#### 2.2.2. Case 2: Increase bandwidth by rerouting

In this case both the routing of demands and allocations are changed. In each iteration (after BSA or ASA) saturated links are removed and all paths using these links are de-allocated. The link capacities should be decreased by the allocated capacity of removed demands ($b_d$). Now the whole algorithm should be run on the reduced graph until there are no more demands. This method has the longest running time, however, it gives the best resource utilisation. It is to be noticed that even in this case the global optimum is not guaranteed. This is because the optimal solution of BSA or ASA is not unique, and the choice of the optimal solution of BSA or ASA may influence the further development of the algorithm and its final results [4].

### *2.3. The weighted RF path and bandwidth allocation*

If we want to prioritise some demands $d$ then a weight factor $w_d$ should be used. By setting $w_1 = 2w_2$ the rate allocated to demand 2 will be increased by double of the increment of demand 1.
In this case everything defined previously is valid, except that in the UMCF subroutine we should add the weight factor $w_d$ for each demand $d$ to the Eq. (6), as follows:

$$b_d = m_d + w_d\alpha^{(k)}(M_d - m_d)\,. \tag{12}$$

In Step 7 in Section 2.2.1 (Eq. (11)) the same should be done, and (10) (and analogously (9)) should be extended to:

$$\alpha^{(k)} = \min_l \left\{ \frac{Y_l^{(k-1)} - \sum_d m_d x_l^d}{\sum_d w_d x_l^d (M_d - m_d)} \right\}. \tag{13}$$

If we want to increase the network throughput, we can prioritise those demands which use shorter paths by setting $w_d$ to be equal to the reciprocal value of the length of the demand, where the length is expressed in number of hops along the shortest possible path between the end-nodes of that demand. This leads to similar fairness definition than PRF. Further on we will deal with the weighted case only, assuming $w_d = 1$, $\forall d = 1,2,...,D$ for the unweighted case.

## 3. Max-min fairness: formulation and algorithms

First, we consider the case without bounds on the demand bandwidths, i.e., we will assume that an infinite amount of traffic is to be carried between the node-pairs. The task is to find optimal paths that allow the highest throughput, while giving the same chance to all demands, i.e., guaranteeing fairness.
Here, instead of parameter $\alpha$, parameter $\beta$ will be used with slightly different meaning as follows. $\beta$ stands for capacity allocated to demands. In this section it will be equal for all demands $d = 1,2,...,D$.

Objective:

$$\max \quad \beta\,. \tag{14}$$

Subject to constraints (4), (5) and:

$$\beta \sum_d x_l^d \le C_l \quad l = 1,2,...,L\,. \tag{15}$$

### *3.1. Algorithms for a single* $\boldsymbol{\beta}$ *for the whole network*

Here the UMCF algorithm described in Section 2.1.2 has to be changed only, as follows.

#### 3.1.1. The UMCF2 subroutine

This subroutine finds the optimal routing for fixed $\beta$. If it had not been fixed, this would have been the exact formulation where $\beta$ and the paths are optimised simultaneously, however, then the problem would have been nonlinear. The difference to UMCF is that $b_d = \beta$, $d = 1,2,...,D$ should be used instead of (6).
Note, that if $\beta$ is a constant it can be avoided in the objective function.

Set:

$$b_d = \beta \quad d = 1,2,...,D\,. \tag{16}$$

Objective:

$$\min \left\{ \beta \sum_d \sum_l x_l^d \right\}. \tag{17}$$

Subject to constraints (4), (5) and (15).
As mentioned, this subroutine finds the optimal routing for fixed $\beta$. The value of $\beta$ can be set iteratively either by the modified BS algorithm (Section 2.1.1) or by the modified AS algorithm.

#### 3.1.2. Binary search algorithm for MMF

The BSA (Section 2.1.1) should be modified to be used for path optimisation with MMF fairness scenario as follows. The initial value of $\beta$ should be set as follows:

$$\beta = \min_l \frac{C_l}{D}\,. \tag{18}$$

Then the value of $\beta$ is increased iteratively by, e.g., 50–100% ($\beta^{(k)} = 1.5\beta^{(k-1)}$) while the problem can be solved. The value of $\beta$ in the last $k^{th}$ iteration will be the upper bound, while the lower bound will be its value in the $(k-1)^{th}$ iteration. Now, binary search between these two values can be used for finding $\beta$ to required accuracy.

### 3.1.3. Adaptive search algorithm for MMF

The AS algorithm (Section 2.1.3) should be modified to be used for path optimisation with MMF fairness scenario as follows.
In this extension of BSA the idea is to increase $\beta$ without changing the paths. After a feasible UMCF2 subroutine we find a new value of $\beta^{(k+1)}$ to be used in the forthcoming $(k+1)^{th}$ iteration based on the paths of the current $k^{th}$ iteration. The new $\beta$ is calculated by the following equation, derived from constraint (15):

$$\beta^{(k)} = \min_l \left\{ \frac{C_l}{\sum_d x_l^d} \right\}. \tag{19}$$

This increase of parameter $\beta$ is carried out after each feasible UMCF2 subroutine, which speeds up the algorithm or increases the precision of $\beta$.

## *3.2. Allowing slightly different values of $\beta$ within a network (MMF/NB/D)*

Since the rate of all traffic is changed equally according to the definition of parameter $\beta$, the first saturated link will limit value of $\beta$. Therefore, an iterative approach is needed, which increases the network throughput, however, it deteriorates the fairness slightly, by offering more resources to demands not using saturated links. The idea is to set a new value of $\beta^{(k)}$ for yet unsaturated links in each iteration $k$. Now we will have different values of $\beta$ for different demands or different sets of demands. Although this allows different rates to different demands, it does not really deteriorate the fairness, since demands having lower rates would not have been able to use higher rates due to bottlenecks, which can not be avoided (even re-routing does not help).

Note, that there are two alternatives analogously to 2.2.:

*Case 1*: The paths of demands are determined in the first iteration. They are not changed any more, only the bandwidths.

*Case 2*: Both, the paths and bandwidths are improved in each iteration.

### 3.2.1. Case 1: Increase bandwidth

In this case the paths assigned to demands are determined within the first phase and are not changed any more. The allocations are changed only, according to the following algorithm:

*Step 1*: Set $k = 0$, $\beta^{(0)} = \beta$, $Y_l^{(0)} = C_l - \beta^{(0)} \sum_d x_l^d$.

*Step 2*: Set $k$++.

*Step 3*: Remove all saturated links and paths using these links.

*Step 4*: If there is no more demand left or $\beta^{(k-1)} = \beta^{(k-2)}$ then *Stop*, otherwise continue.

*Step 5*:

$$\beta^{(k)} = \min_l \left\{ \frac{Y_l^{(k-1)}}{\sum_d x_l^d} \right\}. \tag{20}$$

*Step 6*:

$$Y_l^{(k)} = Y_l^{(k-1)} - \beta^{(k)} \sum_d x_l^d .$$

*Step 7*: Go to Step 2.

The new $\beta$ is calculated by Eq. (20) that has analogous meaning to Eq. (19).
Note, that this iterative procedure has to be repeated up to $L$ times in total for the considered network, since each iteration will saturate at least one link.

### 3.2.2. Case 2: Increase bandwidth by rerouting

In this case both the routing of demands and allocations are changed. In each iteration saturated links should be removed with all paths using these links. The link capacities should be decreased by the capacity allocated to demands removed (by $\beta$). Now the whole algorithm should be run for the reduced graph.
This method has the longest running time, however, it gives the best resource utilisation. It is to be noticed that the global optimum is not guaranteed for the reasons mentioned in Section 2.2.2.

## *3.3. The weighted MMF path and bandwidth allocation*

In this case everything defined previously in Section 3 is valid, except that everywhere (e.g., in Eq. (15)) $w_d x_l^d$ should be written instead of $x_l^d$ and $\sum_d w_d$ should be written instead of $D$ in Eq. (18). Further on we will deal with the weighted case only, assuming $w_d = 1$, $\forall d = 1,2,...,D$ for the unweighted case.

## *3.4. Max-min fairness with bounds (MMF/B)*

In this subsection we assume that each demand has a lower bound ($m_d$) and an upper bound ($M_d$). The lower bound is taken into account by simply modifying the capacity constraints of RF/B in the following way. $\beta$ should be written instead of $\alpha$, and 1 should be written instead of $(M_d - m_d)$, i.e., $(M_d - m_d)$ should be simply left out from the formulation.
The upper bound is handled by introducing an auxiliary leaf node $v_d$ for each demand $d$ and a new link of capacity $M_d$ from the source node of demand $d$ to $v_d$ and finally

changing the source of $d$ to $v_d$. Another way of handling upper bounds is to introduce extra constraints into the ILP formulations.

## 4. Proportional rate fairness: formulation and algorithms

The above fairness definitions ensure optimal fairness measured either relative to the upper and lower bounds (RF) or in absolute units (MMF). However, in these cases the connections spanning more distant points (more hops) will adapt their rate in the very same way, as those close to each other.
To increase the throughput the fairness criteria should be redefined in manner to prioritise connections having less hops (i.e., using less resources) to those which are more distant. F. Kelly *et al.* have proposed the concept of proportional rate fairness [3] where the objective to be optimised is the sum of logarithms of the capacities used by certain demands (e.g., $b_d$), while the constraints are the same as in our previous formulations.

Objective:

$$\max \quad \sum_d \lg b_d \,. \tag{21}$$

Subject to constraints (4), (5) and:

$$\sum_d x_l^d b_d \le C_l \quad l = 1,2,...,L. \tag{22}$$

Unfortunately, this is a convex problem that is nonlinear. To handle this problem a piece-wise linear approximation of the logarithmic function is applied by introducing an auxiliary variable $f_d$ for each demand $d$ as proposed in [4]. The modified objective will be

$$\max \sum_d f_d \tag{23}$$

and additionally the following constraints are given for each demand $d$:

$$f_d \le r_k b_d + s_k, \quad k = 1,...,K. \tag{24}$$

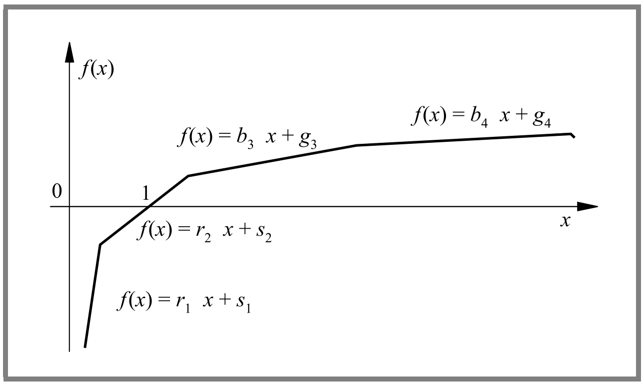


***Fig. 1.*** The piece-wise linear approximation of the logarithmic function.

Figure 1 shows the approximation for $K = 4$ linear pieces, however, in practice more pieces can be used.
In our study the following inequalities were used:

$$f_d \le 4.023595 b_d - 2.704945, \tag{25}$$
$$f_d \le 1.386294 b_d - 1.386294, \tag{26}$$
$$f_d \le 0.693147 b_d - 0.693147, \tag{27}$$
$$f_d \le 0.305430 b_d + 0.082287, \tag{28}$$
$$f_d \le 0.109861 b_d + 1.060132, \tag{29}$$
$$f_d \le 0.034399 b_d + 2.192062. \tag{30}$$

However, after eliminating the logarithmic function from the objective, another problem occurs, namely that constraint (22) is not linear. To avoid this we introduce a new variable $y_l^d$, which represents the flow value of demand $d$ on link $l$. By the following formulation the problem is linear, however it enables split flows.

Objective: (23).

Constraints: (24) and:

$$\sum_{j=1}^{N} y_{ij}^d - \sum_{k=1}^{N} y_{ki}^d = \begin{cases} b_d & \text{if } i \text{ is the source of } d \\ -b_d & \text{if } i \text{ is the sink of } d \\ 0 & \text{otherwise} \end{cases}, \tag{31}$$
$$i = 1,2,...,N, \quad d = 1,2,...,D$$

$$\sum_d y_l^d \le C_l \quad l = 1,2,...,L. \tag{32}$$

To avoid split flows additional constraints are needed and the following final formulation is proposed: $M$ is a large number.

Objective: (23).

Constraints: (4), (5), (31), (24), (32) and:

$$y_l^d \le M x_l^d \quad l = 1,2,...L, \quad d = 1,2,...,D. \tag{33}$$

The bounded case (PRF/B) can be handled by simply introducing constraints into the above ILP formulation.
Although PRF deteriorates fairness in sense of earlier fairness definitions, it increases the throughput.

## 5. Comments and improvements

Although the problems have been defined here for unsplittable flows only, all the methods can be used for splittable flows as well. This even reduces the complexity, since linear programming can be used instead of integer linear programming or mixed integer programming.
When both elastic and rigid traffics coexist in a network, the model has not to be changed, only $m_d$ and $M_d$ values are to be set to be equal ($m_d = M_d$) for all rigid demands. However, if the problem is being solved by a mixed integer linear programming (MILP) solver it might be useful to introduce new variables instead. This will reduce the number of constraints and it will speed up solving the problem and

allow problems of larger scale to be solved. In the numerical results we will deal with elastic traffic only, since as shown this does not reduce the generality.
We believe that the proposed approach ensures the highest fairness, i.e., RF is more fair than the plain MMF. Furthermore, the formulation of the joint path and bandwidth optimisation guarantees higher (or at least equal) throughput than the one with pre-defined paths.

### *5.1. Iterative elastic simulated allocation (IESA)*

The methods spend most of their time in the UMCF or UMCF2 subroutine. The ILP formulation of them contains $LD$ variables and $ND+L$ constraints. Several methods have been proposed in the literature that solve the unsplittable multicommodity flow problem much faster, e.g., SA++ or CA++ in [10]. We have applied SA++ in this study that is based on simulated allocation. The main idea behind simulated allocation [11] is a randomised alternative path allocation and de-allocation of the traffic demands.
Using of SA++ is proposed in larger networks (e.g., with more than 15 nodes), which does not guarantees the optimal solution, but is much faster than the method based on ILP.

Using ILP in the UMCF subroutine is called elastic ILP (EILP), while replacing the UMCF subroutine with SA++ is called iterative elastic simulated allocation.

### *5.2. Elastic simulated allocation (ESA)*

Simulated allocation can be used in a more sophisticated manner as well. The main point of this improvement is that after several iterations of allocations and de-allocations a special procedure called *bandwidth tuner* is called. The bandwidth tuner procedure tunes (changes) the bandwidth of each demand according to the appropriate fairness definition. For example, in case of RF it decreases the value of $\alpha$ if any demand can not be allocated, or increases the value of $\alpha$ if all demands can be allocated and more free space is available in the network. This method is called elastic simulated allocation.

### *5.3. Iterative heuristic for PRF (IPRF)*

The ILP formulation of the PRF definition is very complex: it contains $2(L+1)D$ variables and $2ND+KD+L$ constraints. In order to speed up the calculation the following iterative heuristic method is proposed:

*Step 0*: Find a feasible system of paths by applying MMF/NB/U or MMF/B/U. Set $k=0$ and $\gamma^{(0)}=-\infty$.

*Step 1*: Increase $k$ and find the bandwidth $b_d$ for each demand $d$, according to PRF definition by solving the above problem (that is linear in this case) by an ILP solver. Let $\gamma^{(k)}$ the objective value of the problem. If $\gamma^{(k)}$ has been increased ($\gamma^{(k)}>\gamma^{(k-1)}$) then continue, otherwise *Stop*.

*Step 2*: Run UMCF with bandwidths ($b_d$s) found in Step 1. Go to Step 1.

### *5.4. Shortest paths algorithm (SPA)*

A simple method called shortest paths algorithm has been also implemented. It finds a shortest path for each demand and sets the bandwidth of the demand according to the appropriate fairness definition. This method is similar to those previous methods that assume fixed paths, i.e., it is not able to change the path only the bandwidth of the demands.

## 6. Numerical results

The tests have been carried out on six networks with different number of nodes and links (Table 1). The bounds of traffic demands have been chosen randomly so that the task was not trivial, i.e., using $m_d$ parameters they fit into capacities, while with $M_d$ not.

Table 1
Details of the six test networks

| Details | N5 | N5A | N12 | N15 | N25 | N35 |
|---|---|---|---|---|---|---|
| Nodes | 5 | 5 | 12 | 15 | 25 | 35 |
| Links | 5 | 6 | 18 | 15 | 31 | 51 |
| Demands | 10 | 10 | 66 | 105 | 300 | 595 |

The methods have been compared according to 4 groups of criteria: computational time, network throughput, fairness parameters and hop number. The network throughput (TP) is expressed as the total of carried traffic for all demands. Fairness parameters are min($\alpha$), av($\alpha$), min($\beta$), av($\beta$) and $\gamma$ as defined in Section 1. Average and maximal hop number, av(H) and max(H), indicate the average and maximal hops used by the system of paths.
The results are summarised in Table 2 for methods EILP, IESA and SPA on N12 which represents a relevant part of the Polish backbone. Considering running time, both EILP and IESA is about 12 times faster in case of RF/B than in case of MMF/B. The reason for this is that the addition of $D$ new links and nodes increases the running time significantly. IESA (the heuristic method) is about an order faster than EILP. IESA yields a little worse result than EILP, but still much better then SPA in sense of throughput and fairness parameters. However, average and maximal hop numbers are higher since randomised heuristic allows longer paths. From these results it can be stated that joint path and bandwidth allocation yields better results in sense of throughput and fairness.
It is interesting to compare the fairness parameters (min($\alpha$), av($\alpha$), min($\beta$), av($\beta$), $\gamma$) according to the fairness that had been considered in the optimisation phase. For example, in case of RF min($\alpha$) and av($\alpha$) are relatively high compared to MMF and PRF, however, it yields lower values for min($\beta$), av($\beta$) and $\gamma$.
Considering PRF this yields the highest throughput, $\gamma$ and also av($\beta$), and not significantly worse min($\beta$). Consequently, this seems to be very promising in the unbounded case. However, in the bounded case it gives very poor

Table 2

Numerical results of methods EILP, IESA and SPA for the N12 network

| **EILP** | | Time | TP | min($\alpha$) | av($\alpha$) | min($\beta$) | av($\beta$) | $\gamma$ | av(H) | max(H) |
|---|---|---|---|---|---|---|---|---|---|---|
| RF/B | BS | 34.1 | 104.9 | 0.083 | 0.083 | 1.249 | 1.590 | 27.153 | 2.20 | 5 |
| | AS | 16.2 | 105.0 | 0.083 | 0.083 | 1.250 | 1.591 | 27.204 | 2.20 | 5 |
| | *Case 1* | 16.2 | 151.7 | 0.083 | 0.268 | 1.250 | 2.298 | 45.877 | 2.20 | 5 |
| | *Case 2* | 39.4 | 152.7 | 0.083 | 0.276 | 1.250 | 2.314 | 46.979 | 2.23 | 5 |
| MMF/B | BS | 293.9 | 105.7 | 0.055 | 0.095 | 1.328 | 1.601 | 28.832 | 2.24 | 5 |
| | AS | 305.1 | 106.0 | 0.056 | 0.096 | 1.333 | 1.606 | 29.060 | 2.24 | 5 |
| | *Case 1* | 308.0 | 145.3 | 0.056 | 0.256 | 1.333 | 2.201 | 43.831 | 2.24 | 5 |
| | *Case 2* | 473.4 | 144.9 | 0.056 | 0.259 | 1.333 | 2.195 | 44.523 | 2.35 | 6 |
| PRF/B | | 56.4 | 165.0 | 0.000 | 0.367 | 1.000 | 2.500 | 51.698 | 2.39 | 5 |
| MMF/NB | BS | 15.8 | 92.4 | –0.100 | 0.070 | 1.400 | 1.400 | 22.204 | 2.20 | 5 |
| | AS | 15.6 | 92.4 | –0.100 | 0.070 | 1.400 | 1.400 | 22.207 | 2.20 | 5 |
| | *Case 1* | 15.7 | 236.9 | –0.100 | 0.660 | 1.400 | 3.590 | 56.391 | 2.20 | 5 |
| | *Case 2* | 42.6 | 230.6 | –0.100 | 0.606 | 1.400 | 3.494 | 56.610 | 2.24 | 5 |
| PRF/NB | | 47.9 | 243.0 | 0.015 | 0.342 | 1.000 | 3.682 | 59.826 | 2.26 | 5 |

| **IESA** | | Time | TP | min($\alpha$) | av($\alpha$) | min($\beta$) | av($\beta$) | $\gamma$ | av(H) | max(H) |
|---|---|---|---|---|---|---|---|---|---|---|
| RF/B | BS | 1.4 | 94.6 | 0.042 | 0.042 | 1.126 | 1.433 | 20.308 | 2.42 | 5 |
| | AS | 1.5 | 100.8 | 0.067 | 0.067 | 1.200 | 1.527 | 24.510 | 2.50 | 8 |
| | *Case 1* | 1.7 | 141.7 | 0.067 | 0.235 | 1.200 | 2.148 | 41.193 | 2.50 | 7 |
| | *Case 2* | 4.0 | 144.9 | 0.067 | 0.232 | 1.200 | 2.196 | 41.332 | 2.68 | 7 |
| MMF/B | BS | 43.3 | 101.0 | 0.043 | 0.074 | 1.258 | 1.531 | 25.669 | 2.59 | 5 |
| | AS | 31.6 | 89.5 | 0.014 | 0.024 | 1.083 | 1.356 | 17.054 | 2.41 | 8 |
| | *Case 1* | 31.8 | 135.8 | 0.012 | 0.212 | 1.071 | 2.058 | 36.053 | 2.62 | 9 |
| | *Case 2* | 155.4 | 137.9 | 0.030 | 0.218 | 1.182 | 2.089 | 39.704 | 2.83 | 7 |
| PRF/B | | 0.7 | 146.0 | 0.000 | 0.260 | 1.000 | 2.212 | 38.059 | 2.26 | 5 |
| MMF/NB | BS | 0.7 | 84.8 | –0.119 | 0.037 | 1.286 | 1.286 | 16.578 | 2.35 | 5 |
| | AS | 1.8 | 92.4 | –0.100 | 0.070 | 1.400 | 1.400 | 22.207 | 2.61 | 8 |
| | *Case 1* | 1.0 | 225.4 | –0.111 | 0.611 | 1.333 | 3.415 | 46.918 | 2.55 | 7 |
| | *Case 2* | 5.3 | 213.4 | –0.100 | 0.566 | 1.400 | 3.233 | 49.946 | 2.56 | 6 |
| PRF/NB | | 2.4 | 239.0 | –0.167 | 0.659 | 1.000 | 3.621 | 51.902 | 2.52 | 8 |

| **SPA** | | Time | TP | min($\alpha$) | av($\alpha$) | min($\beta$) | av($\beta$) | $\gamma$ | av(H) | max(H) |
|---|---|---|---|---|---|---|---|---|---|---|
| RF/B | U | 0.02 | 28.1 | 0.0165 | 0.0165 | 0.0495 | 0.4263 | –78.5047 | 2.14 | 4 |
| | D | 0.08 | 205.2 | 0.0165 | 0.1739 | 0.1155 | 3.1098 | 23.3567 | 2.14 | 4 |
| MMF/NB | U | 0.02 | 25.4 | 0.003 | 0.0288 | 0.3846 | 0.3846 | –63.0638 | 2.14 | 4 |
| | D | 0.05 | 280.8 | 0.0036 | 0.4563 | 0.3846 | 4.255 | 38.3519 | 2.14 | 4 |
| MMF/B | U | 0.5 | 25.4 | 0.003 | 0.0288 | 0.3846 | 0.3846 | –63.0638 | 2.14 | 4 |
| | D | 0.852 | 206 | 0.0036 | 0.192 | 0.3846 | 3.1212 | 32.6039 | 2.14 | 4 |
| PRF/NB | | 0.09 | 297.2 | 0.0037 | 0.4967 | 0.1 | 4.503 | 37.2381 | 2.14 | 4 |
| PRF/B | | 0.1 | 215.7 | 0.0037 | 0.2052 | 0.1 | 3.2682 | 31.3147 | 2.14 | 4 |

values for min($\alpha$), i.e., if RF notion is assumed to be fair, than many connections come to grief if optimised with PRF. In the bounded case RF/B (Case 2) is better in running time, throughput, hop numbers, and we believe that it is more fair than the simple, unweighted MMF. Summarised, PRF is proposed in the unbounded case, while RF in the bounded case.

In Fig. 2 min($\alpha$) of the four methods are compared. Min($\alpha$) depends on the traffic pattern, i.e., the results can only be compared within one network. EILP yields the best solution while IESA and ESA are also very close to the optimum. ESA is closer to the optimum especially in larger networks. This is a very promising heuristic method for other fairness definitions as well. EILP did not found solution in N25 and N35 in acceptable time. SPA could not solve the problem in N12 and N25, since it works with fixed shortest paths and in these cases the paths violates the capacity constraints even with the lower bounds.

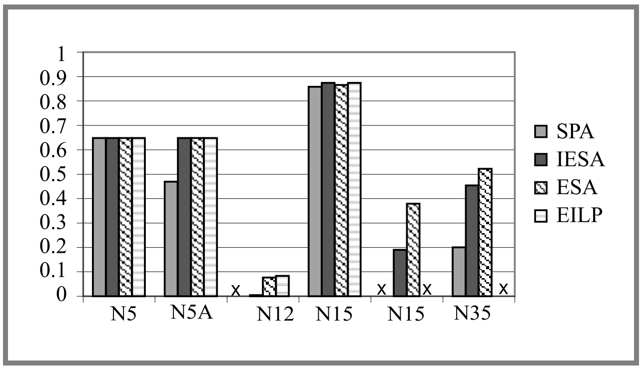


***Fig. 2.*** Min($\alpha$) for the six test networks using algorithms SPA, IESA, ESA and ILP.

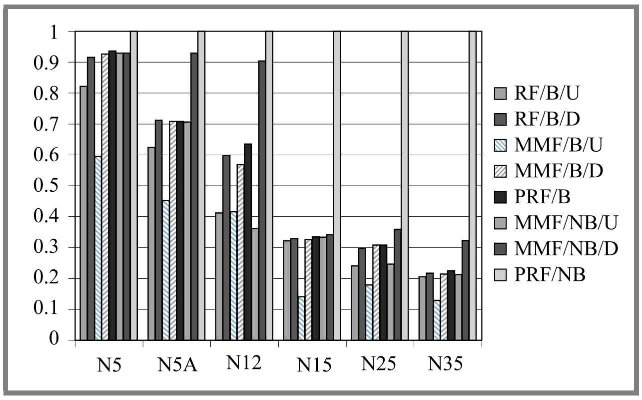


***Fig. 3.*** Throughput of six test networks assuming eight fairness definitions.

In Fig. 3 the throughput is normalised to PRF/NB for each network. Trivially the unbounded (NB) cases always yield higher throughput than the bounded (B) case, and the case allowing different (D) parameter yields higher throughput than the case with uniform (U) parameters. RF and MMF have similar throughput, while PRF has higher throughput, especially in the unbounded case. The efficiency of PRF is very convincing in larger networks, since in this case longer paths obtain significantly less bandwidth that makes space for many short paths.

Table 3
Computational time of ILP and IESA for six test networks and five fairness definitions

| **EILP** | N5 | N5A | N12 | N15 | N25 | N35 |
|---|---|---|---|---|---|---|
| RF/B | 0.18 | 0.65 | 39.4 | 8873 | - | - |
| MMF/B | 0.96 | 1.64 | 473 | - | - | - |
| PRF/B | 0.19 | 0.25 | 56.4 | 80.5 | - | - |
| MMF/NB | 0.09 | 0.18 | 42.6 | 30.6 | - | - |
| PRF/NB | 0.14 | 0.22 | 47.9 | 38.3 | - | - |
| **IESA** | N5 | N5A | N12 | N15 | N25 | N35 |
| RF/B | 0.03 | 0.03 | 4.0 | 2.5 | 36.7 | 85.6 |
| MMF/B | 0.15 | 0.15 | 155 | 123 | 3768 | 30240 |
| PRF/B | 0.01 | 0.01 | 0.7 | 0.3 | 2.2 | 8.7 |
| MMF/NB | 0.02 | 0.02 | 5.3 | 2.8 | 60.6 | 91.7 |
| PRF/NB | 0.01 | 0.01 | 2.4 | 3.8 | 27.1 | 100 |

The computational time of EILP and IESA for five fairness definitions is compared in Table 3. In case of EILP it was acceptable only in networks having up to 15 nodes. IESA is faster, however in case of MMF/B further speed up is required.

# 7. Conclusion

A wide range of algorithms has been proposed, which are all a tradeoff (compromise) between network throughput, fairness and computational time.

In all cases the obtained results were better (in sense of fairness and throughput) than for the case of fixed and predefined alternative paths, however, the running time was longer. Joint optimisation of paths and bandwidths appeared to be always better. We have shown that unused capacities can be further utilised to increase the throughput without deteriorating the fairness in its strict sense. We propose to apply relative fairness notion in the bounded case and proportional rate fairness in the unbounded case. Methods based on ILP are proposed for smaller (less than 20 nodes) networks and iterative heuristics for larger networks.

These methods can be used in any centralised resource management system in the new Internet architecture for configuration of ATM, IP and MPLS networks which will carry elastic traffic.

# Acknowledgements

The authors thank Gábor Fodor (Ericsson Research, Sweden) for initiating discussions and bringing our attention to this area.

T. Cinkler is supported by the OTKA grant D42211 and by the János Bólyai foundation.

P. Laborczi is supported through a European Community Marie Curie Fellowship.

## References


[1] D. Bertesekas and R. Gallagher, *Data Networks*. Englewood Cliffs, NJ: Prentice Hall, 1987.

[2] A. Charny, D. Clark, and R. Jain, "Congestion control with explicite rate indication", in *IEEE ICC'95*, Seattle, USA, 1995.

[3] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan, "Rate control for communications networks: shadow prices, proportional fairness and stability", *J. Oper. Res. Soc.*, vol. 49, pp. 237–252, 1998.

[4] G. Fodor, G. Malicskó, M. Pióro, and T. Szymański, "Path optimization for elastic traffic under fairness constraints", ITC17, Salvador da Bahia, Brazil, 2001.

[5] L. Massouline and J. W. Roberts, "Bandwidth sharing: objectives and algorithms", in *IEEE INFOCOM'99*, New York, USA, 1999.

[6] Y. T. Hou, H. H.-Y. Tzeng, and S. S. Panwar, "A simple ABR switch algorithm for the weighted max-min fairness policy", in *IEEE ATM Worksh.'97*, Lisboa, Portugal, 1997.

[7] H. Qingyanga and D. W. Petr, "Global max-min fairness guarantee for ABR flow control", in *IEEE INFOCOM'98*, San Francisco, USA, 1998.

[8] Q. Ma, P. Steenkiste, and H. Zhang, "Routing high-bandwidth traffic in max-min fair share networks", in *SIGCOMM'97*, Cannes, France, 1997.

[9] S. P. Abraham and A. Kumar, "A new approach for asynchronous distributed rate control for elastic sessions in integrated packet networks", *IEEE/ACM Trans. Netw.*, vol. 9, no. 1, 2001.

[10] P. Laborczi and P. Fige, "Static LSP routing algorithms for MPLS networks", in *Networks 2000*, Toronto, Canada, 2000.

[11] P. Gajownicek, M. Pióro, and A. Arvidsson, "VP reconfiguration through simulated allocation", *NTS*, vol. 13, pp. 251–260, 1996.


**Tibor Cinkler** received his M.Sc. ('94) and Ph.D. ('99) degrees from the Budapest University of Technology and Economics, Hungary, where he is currently Associate Professor at the Department of Telecommunications and Media Informatics. His research interests focus on routing, design, configuration, dimensioning and resilience of IP, MPLS, ATM, NG-SDH and particularly of WR-DWDM based multilayer networks. He has been involved in a few related European and Hungarian projects (ACTS, COST, ETIK, IKTA) and he is member of ONDM, DRCN, Opticomm, EUNICE, ICOCN, etc. Scientific Committees. He is author of over 60 referred scientific publications and of 3 patents.

e-mail: cinkler@tmit.bme.hu
Department Telecommunications and Media Informatics
Budapest University of Technology and Economics
Magyar tudósok krt. 2
H-1117 Budapest, Hungary

**Péter Laborczi** received his B.Sc. and M.Sc. degrees in 1999 and his Ph.D. degree in 2002 in Computer Science from the Budapest University of Technology and Economics in Hungary, Department of Telecommunications and Telematics. Since 2002 he has been a Marie Curie Postdoc research fellow at Arsenal Research in Vienna, Austria. His research fields include configuration, optimization, routing, protection and restoration of infocommunication (mainly MPLS and WDM) networks and similar areas of transport technologies.

e-mail: peter.laborczi@arsenal.ac.at
Arsenal Research
Business Area of Transport Technologies
Faradaygasse 3, Objekt 219C
1030 Vienna, Austria

**Michał Pióro** is a Professor at the Institute of Telecommunications, Warsaw University of Technology and at the Department of Communication Systems, Lund University. He received a Ph.D. degree in telecommunications in 1979 and a D.Sc. degree in 1990, both from the Warsaw University of Technology. In 2002 he received a Polish State Professorship. During 1984–87 and 1997 he had been with the Lund Institute of Technology, leading research projects for Ericsson Telecom AB in network design. In 1986, 2001 and 2003 he served as a senior expert for ITU. During 1990-1991 he had been working for Alcatel SESA, Spain, as a consultant in dynamic routing strategies. In Fall'2002 semester he was a visiting scholar and Professor at University of Missouri-Kansas City, USA. Prof. Pióro wrote three monographs and over 100 papers presented in international telecommunications journals and conference proceedings. He is a Technical Editor of *IEEE Communications Magazine* and a member of program committees of many important telecommunication conferences. His research interests concentrate on modelling, design and performance evaluation of telecommunications networks.

e-mail: mpp@tele.pw.edu.pl
Institute of Telecommunications
Warsaw University of Technology
Nowowiejska st 15/19
00-665 Warsaw, Poland