

# A Nature-Inspired Algorithm for Intelligent Optimization of Network Resources

Xiang Feng

Department of Computer Science  
East China University of  
Science and Technology  
Shanghai, China 200237

Email: xfeng{@ecust.edu.cn, @cs.hku.hk}

Francis C.M. Lau

Department of Computer Science  
The University of  
Hong Kong, Hong Kong  
Email: fcmlau@cs.hku.hk

Dianxun Shuai

Department of Computer Science  
East China University of  
Science and Technology  
Shanghai, China 200237

Email: shdx411022@online.sh.cn

**Abstract**—In complex computer networks having the characteristic of social dynamics, bandwidth allocation is a fundamental problem where bandwidth has to be reserved for connections in advance. This paper presents the theory and approach of the *economic generalized particle model* (EGPM) for intelligent allocation of network bandwidth. This approach transforms the complicated network bandwidth allocation problem into efficient, parallel allocation of network bandwidth. This approach is an important extension and further development of the generalized particle model (GPM) [1]. EGPM emphasizes the use of pricing as the network control mechanism. For the pricing, it makes use of the *tatonnement process* in economics. EGPM arises from GPM but can overcome some of GPM's deficiencies for the network bandwidth allocation problem.

**Index Terms**—Intelligent bandwidth allocation, economic generalized particle model (EGPM), price and demands dynamic modulation, distributed and parallel algorithm, dynamical process.

## I. INTRODUCTION

This paper studies pricing as a network control mechanism. Such a framework has received significant interest in the literature (e.g., see [2]–[6] and the references therein), wherein price provides a good control signal because it carries monetary incentives. The network can use the current price of a resource as a feedback signal to coerce the users into modifying their actions (e.g., changing the rate or route).

## II. PROBLEM MODEL FOR BANDWIDTH ALLOCATION

We apply the *economic generalized particle model* (EGPM) to solve the bandwidth allocation problem for networks that support Quality of Service (QoS), such as ATM, IP Integrated Services, and IP Differentiated Services. The important issues to consider in any practical solution to the problem include: 1) the efficiency of allocation, while taking into account the different needs and performance requirements of the nodes involved; 2) the notion of fairness; 3) the ability to implement the allocation scheme in a distributed manner with minimal communication overheads; and 4) the pricing of the bandwidth so as to maximize the network's revenue. EGPM purports to address all these issues.

In the following, a source-destination pair communicate through a channel associated with the pair, which has a certain demand for bandwidth. The channel can be realized

through one or more paths into which the channel's traffic would be split. We examine the best ways to allocate the bandwidth of a physical link among competing traffics. We consider the scenario with a given number of links, a given number of channels (source-destination pairs) having different demands for bandwidth, and a given number of paths per each of these channels. This is the case of a static environment. Our approach applies also to the dynamic case where the environment changes over time.

The network we study is modeled by a undirected graph. Each channel (corresponding to a pair of source and destination) is split into a number of paths,  $n_j$  ( $j = \overline{1, J}$ ), where  $J$  is the number of channels. The bandwidth allocation problem for a network of  $N$  nodes,  $m$  links,  $J$  channels, and  $n_j$  paths per channel is described by Table 1. We use the logical variable  $y_{uv}$  to express the connectivity of the network:  $y_{uv} = 1$  ( $u, v = \overline{1, N}$ ) if there is a link between nodes  $u$  and  $v$ ; 0 otherwise. In fact,  $m = \sum_{u=1}^N \sum_{v=u+1}^N y_{uv}$ . The notations in

Table 1 are explained below.

$A_i$	The $i$ -th physical link ( $i = \overline{1, m}$ ).
$T^{(j)}$	The $j$ -th channel ( $j = \overline{1, J}$ ; $J \leq N \times (N - 1)/2$ ).
$n_j$	The number of the paths of channel $T^{(j)}$ ( $n = \sum_{j=1}^J n_j$ ).
$T_k^{(j)}$	The $k$ -th path of the $j$ -th channel.
$r_i$	The maximum bandwidth of link $A_i$ .
$d^{(j)}$	The bandwidth demand of channel $T^{(j)}$ .
$x_{ik}^{(j)}$	A logical variable: $x_{ik}^{(j)} = 1$ implies that path $T_k^{(j)}$ passes through link $A_i$ , otherwise 0.
$a_{ik}^{(j)}$	The bandwidth of link $A_i$ that is allotted to path $T_k^{(j)}$ .
$p_k^{(j)}$	The price per unit bandwidth that path $T_k^{(j)}$ tries to pay.
$a_k^{(j)}$	The actual bandwidth obtained by path $T_k^{(j)}$ —that is, $a_k^{(j)} = \min_i \{a_{ik}^{(j)} \mid \forall x_{ik}^{(j)} = 1\}$ (assuming that if $x_{ik}^{(j)} = 0$ , then $a_{ik}^{(j)} = 0$ ).

## III. THE EGPM ARCHITECTURE

Our EGPM-based approach and its corresponding algorithm are divided into two parts: (1) dynamic allocation of network

Table 1 The bandwidth allocation problem

Path \ Link	$T_1^{(1)}$	...	$T_{n_1}^{(1)}$	...	$T_1^{(j)}$	...	$T_{n_j}^{(j)}$	...	Link bandwidth
$A_1$	$x_{11}^{(1)}, a_{11}^{(1)}$	...	$x_{1,n_1}^{(1)}, a_{1,n_1}^{(1)}$	...	$x_{11}^{(j)}, a_{11}^{(j)}$	...	$x_{1,n_j}^{(j)}, a_{1,n_j}^{(j)}$	...	$r_1$
$\vdots$	$\vdots$		$\vdots$		$\vdots$		$\vdots$		$\vdots$
$A_i$	$x_{i1}^{(1)}, a_{i1}^{(1)}$	...	$x_{i,n_1}^{(1)}, a_{i,n_1}^{(1)}$	...	$x_{i1}^{(j)}, a_{i1}^{(j)}$	...	$x_{i,n_j}^{(j)}, a_{i,n_j}^{(j)}$	...	$r_i$
$\vdots$	$\vdots$		$\vdots$		$\vdots$		$\vdots$		$\vdots$
$A_m$	$x_{m1}^{(1)}, a_{m1}^{(1)}$	...	$x_{m,n_1}^{(1)}, a_{m,n_1}^{(1)}$	...	$x_{m1}^{(j)}, a_{m1}^{(j)}$	...	$x_{m,n_j}^{(j)}, a_{m,n_j}^{(j)}$	...	$r_m$
Allotted bandwidth (price) to paths	$p_1^{(1)}, a_1^{(1)}$	...	$p_{n_1}^{(1)}, a_{n_1}^{(1)}$	...	$p_1^{(j)}, a_1^{(j)}$	...	$p_{n_j}^{(j)}, a_{n_j}^{(j)}$	...	
Required bandwidth of channels	$d^{(1)}$			...	$d^{(j)}$			...	

bandwidth based on GPM, and (2) dynamic modulation of price and demands for network bandwidth.

GPM uses a resource force-field to model the interaction of the network entities [1]. A similar force-field is used here, which consists of numerous particles and forces. A particle having its own dynamic equations represents a network entity, and each force having its own time-varying property represents some social interaction among network entities. The process is iterative, where at every step the allocation module would produce an allocation based on which the other module would generate a new modulated price and set of demands. The iteration terminates when an equilibrium is reached. The two modules can be implemented by two separate algorithms; in this paper, we combine them into a single EGPM algorithm.

#### IV. THE EGPM ALGORITHM

##### A. Allocation model

The allocation policy takes on new parameter values at every step of the iterative process to produce an allocation which is closer to equilibrium than the previous allocations. Since the evolution of the allocation policy in GPM and that in EGPM are almost the same, only the relevant definitions and properties are given here. For further details, we refer the reader to [1].

The mathematical model that involves  $m$  links and  $n$  paths is defined as follows. Note that a particle in the force-field that represents a link in the network is called a link particle, and so on.

Let  $u_i(t)$  be the distance from the current position of link particle  $A_i$  to the upper boundary of the force-field at time  $t$ , and let  $J(t)$  be the total utility of all particles. We define

$$u_i(t) = \exp \left[ - \sum_{j=1}^J \sum_{k=1}^{n_j} p_k^{(j)}(t) \cdot a_{ik}^{(j)}(t) \right]; \quad (1)$$

$$J(t) = \sum_{i=1}^m u_i(t) \quad (2)$$

At time  $t$ , the potential energy function  $P(t)$ , which is caused by the upward gravitational force of the force-field, is defined by

$$P(t) = \varepsilon^2 \ln \sum_{i=1}^m \exp[-u_i^2(t) / 2\varepsilon^2] \quad (3)$$

where  $0 < \varepsilon < 1$  is a parameter to be tuned in the implementation.

At time  $t$ , the potential interaction energy function,  $Q(t)$ , is defined by

$$Q(t) = \beta_1 \sum_{i=1}^m \left| \sum_{j=1}^J \sum_{k=1}^{n_j} a_{ik}^{(j)}(t) - r_i(t) \right|^2 + \beta_2 \sum_{j=1}^J \left| \sum_{k=1}^{n_j} a_k^{(j)}(t) - d^{(j)}(t) \right|^2 \quad (4)$$

where  $0 < \beta_1, \beta_2 < 1$ . The smaller  $Q(t)$  is, the better.  $\beta_1, \beta_2$  are weights applied to the availability of link bandwidth and the satisfactory ratio (to be defined later) of the path demands, respectively.

Eq. (4) describes the effect of interactions among links during the bandwidth allocation process. The first term and the second term of  $Q(t)$  perform penalty functions with respect to the constraints on the utilization of resources (i.e., links) and the degree of satisfaction of the users (i.e., channels) respectively. Therefore, resource utilization and users' satisfaction can be explicitly included as optimization objectives through some appropriate choices of the coefficients  $\beta_1$  and  $\beta_2$  respectively. We suppose that there are specific interactive forces among particles, and these forces may cause the potential energy components represented by the first and second terms of  $Q(t)$  to decrease.

The dynamics of particle  $A_i$  are defined by

$$\Delta a_{ik}^{(j)}(t) = -\lambda_1 \frac{\partial u_i(t)}{\partial a_{ik}^{(j)}(t)} - \lambda_2 \frac{\partial J(t)}{\partial a_{ik}^{(j)}(t)} - \lambda_3 \frac{\partial P(t)}{\partial a_{ik}^{(j)}(t)} - \lambda_4 \frac{\partial Q(t)}{\partial a_{ik}^{(j)}(t)} \quad (5)$$

where  $\lambda_1, \lambda_2, \lambda_3$  are one-dimensional coefficient vectors, and  $\lambda_4$  is two-dimensional coefficient vector.  $\lambda_1 = (\lambda_i^{(1)})_{m \times 1}$ , where  $\lambda_i^{(1)}$  represents the strength for  $A_i$  to pursue personal profit,  $0 \leq \lambda_i^{(1)} \leq 1$ .  $\lambda_2 = (\lambda_i^{(2)})_{m \times 1}$ , where  $\lambda_i^{(2)}$  represents the strength for  $A_i$  to take into account the whole profit of all link particles,  $0 \leq \lambda_i^{(2)} \leq 1$ .  $\lambda_3 = (\lambda_i^{(3)})_{m \times 1}$ , where  $\lambda_i^{(3)}$  represents the strength for  $A_i$  to increase the minimal personal profit among all link particles,  $0 \leq \lambda_i^{(3)} \leq 1$ .  $\lambda_4 = (\lambda_{iu}^{(4)})_{m \times m}$ , where  $\lambda_{iu}^{(4)}$  represents the strength for  $A_i$  to interact with  $A_u$ ,  $0 \leq \lambda_{iu}^{(4)} \leq 1$ .

$A_i$  updates  $a_{ik}^{(j)}$  according to Eq. (5)—that is,  $a_{ik}^{(j)}(t+1) = a_{ik}^{(j)}(t) + \Delta a_{ik}^{(j)}(t)$ . Propositions 2–5 explain the terms in Eq. (5).

We can therefore obtain the radial velocity of link particle  $A_i$  along a vertical direction to the upper boundary of gravitational field by the equation

$$v_i = du_i/dt = \sum_{j=1}^J \sum_{k=1}^{n_j} \frac{\partial u_i}{\partial a_{ik}^{(j)}} \frac{da_{ik}^{(j)}}{dt} \quad (6)$$

The following properties

**Proposition 1** *The decrease of the potential energy  $P(t)$  of the gravitational field will result in the increase of the profit of the link whose profit is the minimum among all links.*

**Proposition 2** *The behavior of  $A_i$ , by the third term of  $\Delta a_{ik}^{(j)}$  (see Eq. (5)), will always bring about an increase of the minimum profit obtained by a link, and this increase is directly proportional to the coefficient vector  $\lambda_3$ .*

**Proposition 3** *The behavior of  $A_i$  caused by the first and third terms of  $\Delta a_{ik}^{(j)}$  (see Eq. (5)) will always result in an increase of the personal profit of  $A_i$ , and this increase is proportional to the coefficient vectors  $\lambda_1$  and  $\lambda_3$ .*

**Proposition 4** *The behavior of  $A_i$  caused by the second term of  $\Delta a_{ik}^{(j)}$  (see Eq. (5)) will increase the global utility of the whole network, which is directly proportional to the coefficient vector  $\lambda_2$ .*

**Proposition 5** *The behavior of  $A_i$  caused by the fourth term of  $\Delta a_{ik}^{(j)}$  (see Eq. (5)) will narrow the gap between bandwidth resources and the demand, and the strength is proportional to the coefficient vector  $\lambda_4$ .*

## B. Pricing model

In this subsection, we present the evolution of the pricing policy. It combines with the evolution of the allocation policy discussed in the previous section to form our EGPM-based approach for intelligent allocation of network bandwidth. We examine the evolutionary model that can mathematically describe the dynamic modulation of price and demands of network bandwidth.

### Excess demand function

The price vector  $p^* = p_1^{*(1)}, \dots, p_k^{*(j)}, \dots, p_{n_j}^{*(J)}$ , a solution of the bandwidth allocation problem  $a = a_1^{(1)}, \dots, a_k^{(j)}, \dots, a_{n_j}^{(J)}$  is in equilibrium if and only if

$$(1) \ a \text{ is a feasible solution, and} \\ (2) \ z_j(p^*) = \sum_{k=1}^{n_j} a_k^{(j)} - d^{(j)} = 0. \quad (7)$$

The mapping  $z$  is called the excess demand function and it has the following properties:

(P1)  $z$  is single valued and continuous for all  $p > 0$ .

(P2)  $z$  satisfies Walras' law:  $p \cdot z(p) = 0$  for all  $p > 0$ .

(P3)  $z$  is homogeneous of degree zero:  $z(\alpha p) = z(p)$  for all  $\alpha > 0$ .

(P4) There is a scalar  $\nu < 0$  such that  $z_j(p) > \nu$  for all  $j$  and  $p > 0$ .

(P5) It holds that  $\lim_{p(t) \rightarrow p} \left[ \max_{j,k \in JK_p} z_j(p(t)) \right] = \infty$ , when  $p(t) > 0$ ,  $p \neq 0$  and  $JK_p = \{j, k : p_k^{(j)} = 0\} \neq \emptyset$ .

Homogeneity is an elementary property that an excess demand function has because the channels' price vector stays the same when the budget constraints are multiplied with positive constants. Walras' law and continuity result from the channels' minimization problems when the preferences are strictly concave and locally non-satiated.

(P4) means that all the component functions of  $z$  are bounded from below on  $\mathbb{R}_+^n$ . This property is there because the net supply of bandwidth to the channels cannot exceed the channels' total demand. According to (P5) all the bandwidth of paths is desirable in the sense that when some of them become free, the excess demand becomes infinitely large at least for a little of the bandwidth of paths. When  $z$  has the properties (P1)–(P5), the economy has at least a ray of equilibrium prices. See, e.g., [7, Chapter 17] for more about the properties of excess demand functions.

The absolute value of the excess demand function is inversely proportional to the *satisfaction ratio* of the path demands, which is defined by  $\sum_{k=1}^{n_j} a_k^{(j)}(t) / d^{(j)}(t)$ . Our pricing policy in EGPM aims at maximizing the satisfaction ratio of the path demands, and therefore the iterative price modulation processes in EGPM have to satisfy the conditions that:

- 1) the absolute value of the excess demand function will be minimized, and
- 2) the modulation processes will converge.

### Iterative price modulation processes

The dynamic modulation of price in the EGPM algorithm is based on the following consideration. The price will rise when the supply of bandwidth is more than the demands of bandwidth ( $z_j < 0$ ), whereas the price will fall when the supply of bandwidth is less than the demands of bandwidth ( $z_j > 0$ ). The tendency to increase or decrease in price is directly proportional to  $|z_j|$ . In economics, such a dynamic modulation of price is known as the tatonnement process. The continuous time tatonnement process, introduced by Samuelson [8], is described by the differential equation

$$\dot{p}(t) = z(p(t)) \quad (8)$$

where  $\dot{p}(t)$  is the time derivative of  $p(t)$ . This process is usually interpreted as an ‘‘auction’’ run by a fictitious agent, which sets the prices until an equilibrium is reached.

It can be shown that under the following stability condition (C1), the process (8) is globally stable, i.e., it converges to an equilibrium for any positive initial prices.

(C1) there exists  $p^* > 0$  that solves (7) and satisfies  $p^* \cdot z(p) > 0$  for all  $p > 0$  for which  $z(p) \neq 0$ .

This convergence condition (C1) can be interpreted as the weak axiom of revealed preferences between the equilibrium  $p^*$  and any disequilibrium price vector.

The simplest discrete time alternative for the process (8) is the fixed-point iteration

$$p(t+1) = p(t) + z(p(t)) \quad (9)$$

where  $t$  is the iteration index that corresponds to the time instants at which the prices are modulated. Analyzing (9)

instead of (8) suits the bandwidth allocation problem better, for which the price modulation process proceeds over discrete time instants. Unfortunately, (9) is not suitable for solving (7), because some of the prices may become negative during the iteration, for which  $z$  is usually not defined.

One way to obtain non-negative prices is to update  $p_k^{(j)}(t)$  as follows.

$$p_k^{(j)}(t+1) = \max\{0, p_k^{(j)}(t) + \theta \cdot z_j(p_k^{(j)}(t))\} \quad (10)$$

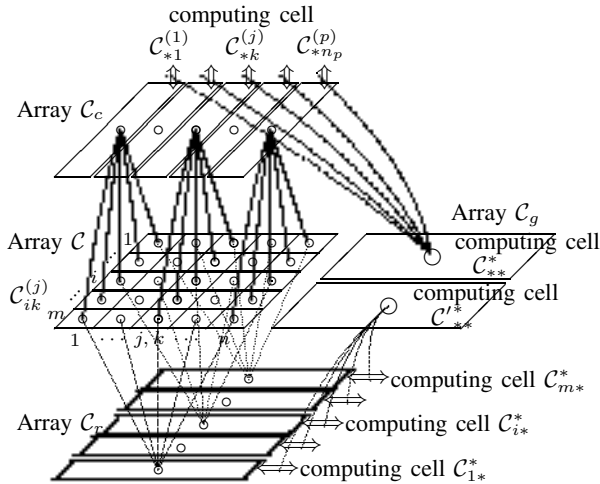
where  $\theta$  is called the price modulation rate which is a positive constant. The convergence of this process restricts the choice of  $\theta$  and  $p(t=0)$ . The drawback of process (10) is that due to (P5) the excess demand function is not finite if some prices become zero. It can be easily seen that the discrete time process (10) does not converge under the same assumptions as the continuous time process (8). For example, the convergence of process (10) depends on the choice of the parameter  $\theta$ . Moreover, normalized discrete time processes tend to exhibit chaotic behavior. We give an alternative discrete time process which converges under assumptions that are very close to those required for the process (8) to converge.

## V. PARALLEL ARCHITECTURE FOR EGPM AND THE EGPM ALGORITHM

### A. Parallel architecture for EGPM

The parallel computing architecture of the EGPM described here is just to demonstrate the possibility of using hardware. It is helpful for understanding the EGPM algorithm and its inherent parallelism. The same functions can be implemented using software.

The parallel computing architecture of the EGPM, as shown in Fig. 1, is composed of four computing cell arrays,  $\mathcal{C}$ ,  $\mathcal{C}_r$ ,  $\mathcal{C}_c$ , and  $\mathcal{C}_g$ , whose computing cells are denoted by  $\mathcal{C}_{ik}^{(j)}$ ,  $\mathcal{C}_{i*}^*$ ,  $\mathcal{C}_{*k}^{(j)}$ ,  $\mathcal{C}_{**}^*$  and  $\mathcal{C}'_{**}$ , respectively.



There is no interconnection between the computing cells in the same array

Fig.1 The parallel computing architecture of the EGPM

The number of computing cells in each array is:  $n \times m$  for  $\mathcal{C}$ ,  $n$  for  $\mathcal{C}_r$ ,  $m$  for  $\mathcal{C}_c$ , and 2 for  $\mathcal{C}_g$ , and hence the

total number of computing cells is  $nm + n + m + 2$ . There is no connection between computing cells in the same array, whereas there are local interconnections between the following computing cell pairs:  $\mathcal{C}_{*k}^{(j)}$  and  $\mathcal{C}_{i*}^*$ ;  $\mathcal{C}_{ik}^{(j)}$  and  $\mathcal{C}_{*k}^{(j)}$ ;  $\mathcal{C}_{i*}^*$  and  $\mathcal{C}'_{**}$ ;  $\mathcal{C}_{*k}^{(j)}$  and  $\mathcal{C}'_{**}$ . It is obvious that the connection degree of each computing cell in the array  $\mathcal{C}$  is at most 2, and the two computing cells in  $\mathcal{C}_g$  have connection degree  $n+m$ . And the total number of interconnections is  $2nm + n + m$ .

At the time  $t$  during a time slot, computing cell  $\mathcal{C}_{ik}^{(j)}$  sends its dynamic state  $a_{ik}^{(j)}(t)$  to computing cells  $\mathcal{C}_{i*}^*$  and  $\mathcal{C}_{*k}^{(j)}$ , and receives feedback inputs that are generated by computing cells  $\mathcal{C}_{i*}^*$  and  $\mathcal{C}_{*k}^{(j)}$  at time  $(t-\tau)$ . Using the received  $a_{ik}^{(j)}(t)$ , computing cell  $\mathcal{C}_{i*}^*$  ( $\mathcal{C}_{*k}^{(j)}$ , resp.) computes its state  $v_i(t)$  ( $p_k^{(j)}(t)$ ,  $z_j(t)$ , resp.) according to Eq. 6 (Eqs. 12, 9, resp.), and sends its current output as feedback to computing cell  $\mathcal{C}_{ik}^{(j)}$ . Meanwhile, computing cell  $\mathcal{C}_{i*}^*$  and  $\mathcal{C}_{*k}^{(j)}$  receive the feedback from computing cell  $\mathcal{C}'_{**}$  and  $\mathcal{C}'_{**}$ . Computing cells  $\mathcal{C}'_{**}$  and  $\mathcal{C}'_{**}$  will change their computed calculation states according to  $z_j$  and  $v_i$ , respectively. Computing cell  $\mathcal{C}_{ik}^{(j)}$  will change its dynamic state according to Eq. 5.

Note that at time  $t$ , in cell array  $\mathcal{C}$ , every computing cell is involved in the particles' dynamics, and the values of the related variables are referred to as "dynamic states". In cell arrays  $\mathcal{C}_c$  and  $\mathcal{C}_r$ , every computing cell evolves based on the "dynamic states" of cell array  $\mathcal{C}$ , and their evolution is captured by a "calculation state".

The implementation of EGPM just illustrated is characterized by high parallelism and good scalability. All the computations of the cellular dynamics, either in the same array or in different arrays, are concurrently carried out. The cellular structure, the computations of the cellular dynamics, and the algorithm are all independent of the scale of the problem in question. Moreover, there is no direct connection between computing cells of the same array, which makes the scheme relatively easy to implement by VLSI.

### B. The EGPM algorithm

Now we present the EGPM algorithm which consists of a component for distributed parallel bandwidth allocation, and a component for distributed asynchronous price-demands modulation (as shown above).

The typical characteristic of network bandwidth allocation is the real-time variation of the supply and demands of bandwidth. In a network, frequently, links are appended to or deleted from; the bandwidth demands between two end nodes arise and end; bandwidth supply or demands increase or decrease. The time quantum  $\Delta t$  used by the iteration process is adjusted and modulated based on the speed of change of the supply and demands of bandwidth.  $\Delta t$  should be made smaller when the supply and demands vary rapidly, and made larger when the supply and demands vary tardily.

The EGPM algorithm is executed to allocate network bandwidth in parallel, by dynamically modulating price and demands of network bandwidth in an asynchronous fashion. Although the solution at equilibrium as worked out by this

## EGPM algorithm

**Input:**  $r_i$

**Output:**

1. Initialization:
  - $t \leftarrow 0$
  - $\Delta t, \theta, \varepsilon, \varsigma$
  - $a_{ik}^{(j)}(t), x_{ik}^{(j)}(t)$  — Initialize in parallel in  $C$  (computing cells  $C_{ik}^{(j)}$ )
  - $p_k^{(j)}(t)$  — Initialize in parallel in  $C_c$  (computing cells  $C_{*k}^{(j)}$ )
2.  $a_k^{(j)}(t) \leftarrow \min_i a_{ik}^{(j)}(t)$  — Compute in parallel in  $C_c$  (involved in  $C_{*k}^{(j)} \mid \forall x_{ik}^{(j)}(t) = 1$ )
- $z_j \leftarrow \sum_{k=1}^{n_j} a_k^{(j)}(t) - d^{(j)}(t)$  — Compute in parallel in  $C_c$  ( $C_{*k}^{(j)}$ )
- $v_i(t)$  — Compute in parallel in  $C_r$  ( $C_{i*}^{(*)}$ ) according to Eq. (6)
  - Judge in  $C_g$  ( $C_{**}^{(*)}$ ) and  $C_{**}^{(*)}$
  - Compute in parallel in  $C_c$  ( $C_{*k}^{(j)}$ )
  - Judge in  $C_c$  ( $C_{*k}^{(j)}$ )
3. **while** ( $|z_j| > \varsigma$  and  $v_i(t) \neq 0$ ) **do**
  - 1)  $p_k^{(j)}(t + \Delta t) \leftarrow p_k^{(j)}(t) + z_j(t) \cdot \theta$  — Compute in parallel in  $C_c$  ( $C_{*k}^{(j)}$ )
  - if** ( $p_k^{(j)}(t + \Delta t) \leq 0$ ) — Judge in  $C_c$  ( $C_{*k}^{(j)}$ )
    - $\theta \leftarrow \theta/2$
    - $p_k^{(j)}(t + \Delta t) \leftarrow p_k^{(j)}(t) + z_j(t) \cdot \theta$  — Compute in parallel in  $C_c$  ( $C_{*k}^{(j)}$ )
    - $d^{(j)}(t + \Delta t) \leftarrow p_k^{(j)}(t) \cdot d^{(j)}(t) / p_k^{(j)}(t + \Delta t)$  — Compute in parallel in  $C_c$  ( $C_{*k}^{(j)}$ )
    - $t \leftarrow t + \Delta t$
  - 2)  $u_i(t)$  — Compute in parallel in  $C_r$  ( $C_{i*}^{(*)}$ ) according to Eq. (1)
    - $v_i(t)$  — Compute in parallel in  $C_r$  ( $C_{i*}^{(*)}$ ) according to Eq. (6)
    - $da_{ik}^{(j)}(t)/dt$  — Compute in parallel in  $C$  ( $C_{ik}^{(j)}$ ) according to Eq. (5)
    - $a_{ik}^{(j)}(t) \leftarrow a_{ik}^{(j)}(t-1) + da_{ik}^{(j)}(t)/dt$  — Compute in parallel in  $C$  ( $C_{ik}^{(j)}$ )
    - if** ( $\sum_{j=1}^l \sum_{k=1}^h a_{ik}^{(j)}(t) \leq r_i$  and  $\sum_{j=1}^l \sum_{k=1}^{h+1} a_{ik}^{(j)}(t) > r_i$ ) — Judge in  $C_r$  ( $C_{i*}^{(*)}$ )
      - $a_{ik}^{(j)}(t) = 0$  — Compute in parallel in  $C$  ( $C_{ik}^{(j)}$ )
      - $a_k^{(j)}(t) \leftarrow \min_i a_{ik}^{(j)}(t)$  — Compute in parallel in  $C_c$  (involved in  $C_{*k}^{(j)} \mid \forall x_{ik}^{(j)}(t) = 1$ )
      - $z_j \leftarrow \sum_{k=1}^{n_j} a_k^{(j)}(t) - d^{(j)}(t)$  — Compute in parallel in  $C_c$  ( $C_{*k}^{(j)}$ )
  - 3) **if** ( $|z_j(t)| \geq |z_j(t - \Delta t)|$ ) — Judge in  $C_c$  ( $C_{*k}^{(j)}$ )
    - $\theta \leftarrow \theta/2$
    - $p_k^{(j)}(t) \leftarrow p_k^{(j)}(t - \Delta t)$  — Compute in parallel in  $C_c$  ( $C_{*k}^{(j)}$ )
    - $d^{(j)}(t) \leftarrow d^{(j)}(t - \Delta t)$  — Compute in parallel in  $C_c$  ( $C_{*k}^{(j)}$ )

$\varepsilon$  is a parameter to be tuned in the implementation,  $\varsigma$  is the desired threshold of  $z_j$ , and  $\theta$  is the price modulation rate. Substeps 1 and 3 of Step 3 implement the price and demands modulation, and Substep 2 of Step 3 implements the bandwidth allocation policy.

asynchronous modulation of price and demands is a local equilibrium solution within a small time frame, it will approach the globally optimal solution within a large time frame. What is more, the asynchronous nature of the algorithm makes it adaptable to complex and fast changing network environments.

## VI. CONCLUSION

We have discussed the theory and approach of the economic generalized particle model (EGPM) for intelligent allocation of network bandwidth. This approach can transform the complicated network bandwidth allocation problem into efficient, parallel allocation of network bandwidth. The mechanism is based on asynchronous modulation of the price of network bandwidth. The proposed model and its algorithm are distinguished by: (1) high parallelism and real-time computational performance; (2) a market-oriented mechanism between the demands and service; (3) the microscopic characterization of an individual link being combined with the macroscopic properties of the whole network, which achieves both fairness and efficiency.

## Acknowledgement

This project is supported in part by the Hong Kong General Research Fund (HKU 713708E).

## REFERENCES

- [1] Dianxun Shuai, Xiang Feng. The Parallel Optimization of Network Bandwidth Allocation Based on Generalized Particle Model. *Computer Networks*. 2006, 50(9): 1219-1246.
- [2] J. K. Mackie-Mason and H. R. Varian, Pricing the Internet. *Public Access to the Internet*, B. Kahin and J. Keller, Eds. Cambridge, MA: MIT Press, 1995, pp. 269-314.
- [3] A. A. Lazar and N. Semret, The progressive second price auction mechanism for network resource sharing. *Proc. 8th Int. Symp. Dynamic Games and Applications*, Maastricht, The Netherlands, Jul. 1998, pp.359-365.
- [4] S. H. Low and D. E. Lapsley, Optimization flow control, I: Basic algorithm and convergence. *IEEE/ACM Trans. Netw.*, 1999, vol. 7, no. 6, pp. 861-874.
- [5] H. Yaiche, R. Mazumdar, and C. Rosenberg, A game theoretic framework for bandwidth allocation and pricing in broadband networks. *IEEE/ACM Trans. Netw.*, 2000, vol. 8, no. 5, pp. 667-678.
- [6] Xiaojun Lin, Ness B. Shroff, Simplification of Network Dynamics in Large Systems. *IEEE/ACM Trans. Netw.*, 2005, vol. 13, no. 4, pp. 813-826.
- [7] Mas-Colell, A., Whinston, M. D., Green, J. R.: *Microeconomic Theory*, New York: Oxford Univ. Press, 1995
- [8] Samuelson, P. A.: *Foundations of Economic Analysis*, Cambridge, MA: Harvard University Press, 1947