

Chapter 2

Network Layer Control System: Consensus-Based Control, Theoretical Results and Performance Issues

2.1 Introduction

During last years control theory approaches have been proposed to deal with the congestion and rate control problem of communication networks. The aim is to avoid congestion at bottleneck nodes by regulating the rate (i.e. allocated bandwidth) of sources traffic (i.e. users or companies requiring network services). Usually *ordinary* and *premium* source traffic-type are considered such that bandwidth is allocated to the first provided a sufficient rate is guaranteed to the latter. The main idea is to use existing protocols (e.g. TCP/IP, ATM) to provide an explicit feedback about the network state for the congestion or rate network control system. The idea of using explicit feedback to perform congestion control has been explored in the wired context and more recently in wireless network environment. Seminal works on ATM network proposed mechanisms for providing explicit-rate control for ordinary available bit rate (ABR) traffic. Then, in the recent years many works have been exploited active queue management (AQM) schemes in TCP/ip-ATM networks to deliver preemptively congestion notification to the source for reducing its transmission rate and therefore avoiding buffer overflow. Design examples of rate controller for ATM networks can be found in [1–4], just to cite a few. Active Queue Management (AQM) strategies to improve the performance of the existing protocols are proposed for wired networks in [5, 6]. Other approaches considered the problem of explicit-rate control [7, 8]. In [9] the authors explore a new congestion control algorithm (e.g., rate control protocol (RCP)) so that a router assigns a single rate to all flows that pass through it. Different control schemes have been proposed for wireless networks (e.g., see [10–13]). The recent increasing diffusion of remote and web application is motivating the development of congestion and rate control schemes in order to enhance the existing wired/wireless protocols to face with the requirement performance of the specific application. Examples can be found in [14–20] where congestion control techniques are proposed to enhance application and transport layer performance of the existing protocols in terms of variable jitter, bandwidth, or congestion.

Among up to date architecture for pervasive web applications, it is worth to mention content delivery network (CDN), that are network aimed to distribute contents to the users [21]. One main aim is leveraging on rate-based scheme to balance the server load [22]. In [21] the authors proposed a hop-by-hop rate control such that an end-to-end quality-rate performance metric is guaranteed in terms of time delay and load balancing.

All the aforementioned approaches concern the design of the rate regulator that uses only local information at the switch or router for control purpose (e.g., queue length, virtual rate, link capacity, incoming traffic estimation), and most of their theoretical results refer to the case of single bottleneck scenarios. In the recent years distributed coordination of multi-agent systems has received significant attention (e.g., see [23] and references therein). One common feature of this research allows every network agent to automatically address a common objective using only local information received from its neighboring agents. While different design techniques for cooperative control of multi-agent systems have been proposed and successfully used in several applications such as formation flight, robot swarm, Lagrangian systems, sensor networks [24–26], renewable energy [27], distributed fault tolerant control [28], the benefits of its application to communication network is quite unexplored. In this direction, recently in [29] it is proposed a controller scheme for DiffServ Network that is partially based on the idea of formation control [30]. Specifically it is used a leader-follower scheme on a link capacity between premium buffer and ordinary buffer at each router. Herein we extensively apply the multi-agent approach to communication network layer by introducing a concept of a cooperation-based rate control in the explicit-rate control framework of communication networks. Herein each router (server or switch) cooperates with its neighbors and regulates the queue length on the base of one-step bottleneck neighbors queue information according to the scheme in Fig. 2.1. This cooperative algorithm operates at the network layer of the multilayer control system presented in Fig. 1.2.

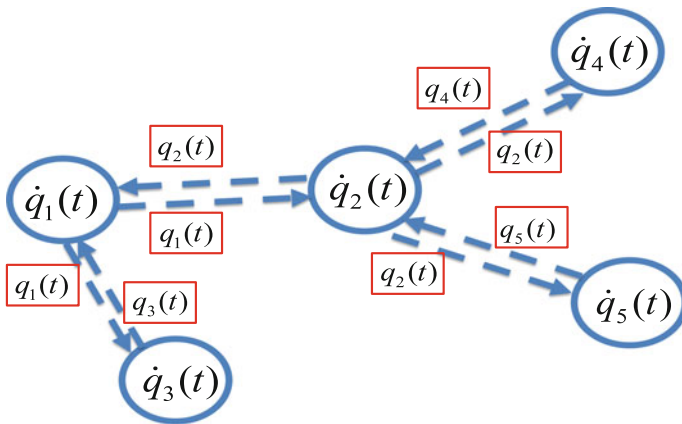


Fig. 2.1 Consensus-based algorithm at the network layer

Specifically we consider a multi-bottleneck model and introduce the concept of “overlay virtual graph” that easily allows to recast the congestion problem in terms of cooperative control. Then, we will present the consensus-based cooperative rate control scheme (in what follows briefly CRC) in order to: (i) stabilize the network and give sufficient stability conditions suitable for network parameters and controller gains design; (ii) guarantee max–min fairly bandwidth allocation to the heterogeneous sources; (iii) balance the network queues length at a given set point value reducing packet loss and improving link utilization. Differently from the standard existing approaches in the literature on the queue balancing, herein the router/server load balancing and network performance are guaranteed without requiring rerouting or hop-by-hop operation. The approach can be easily extended to Diffserv network and applied to different technology or web-based application scenarios (Internet, content delivery network) and therefore it is of interest for the industrial community. Moreover, the presence of multi-bottleneck and heterogeneous sources time delay is taken into account in the problem formulation.

Finally, the proposed algorithm can be implemented both on wired and wireless network technology by respectively adopting end-to-end and hop-by-hop communication mechanism.

2.2 Network Model and Overlay Virtual Graph

Let $\mathbb{C}^n$ the n -dimensional complex vector space, the i th component of a vector $x \in \mathbb{C}^n$ is denoted by x_i , while $\mathbf{1}_n$ is the n -dimensional vector of all 1 and $X = \text{diagonal}\{x\}$ is a diagonal matrix in $\mathbb{C}^{n \times n}$ generated by the vector x and having x as diagonal. Given $z \in \mathbb{C}$, $|z|$ denotes its magnitude value. Given a matrix A , $\sigma(A)$ denotes the spectrum of A while $f(A)$ is the field of values of A . For a set $V \subset \mathbb{C}$, $\text{Co}(V)$ denotes the convex hull of V , while $|V|$ denotes the cardinality of V . For a square matrix B with real eigenvalues, $\lambda(B)$ denotes its spectrum, λ_m (λ_M) denotes the algebraically smallest (largest) eigenvalue, while s represents the Laplace complex variable and j the imaginary unit. Let $G(N, E, A)$ be a graph with the set of nodes N , set of edges $E \subseteq N \times N$, and an adjacency matrix $A = \{a_{ij}\}$ with nonnegative adjacency elements. The set of neighbors of the i th node is defined by $N_i = \{k \in N : a_{ik} = 1\}$. Considering an undirected graph, the degree value d_i of the node i th is the number of the neighbors of the node i th. The Laplacian is a matrix L of elements l_{ij} such that $l_{ij} = \sum_{j=1, j \neq i}^N a_{ij}$ if $i = j$, $l_{ij} = -a_{ij}$, if $i \neq j$. The Laplacian can recast as $L = D - A$ with D is the $N \times N$ diagonal matrix having in position i th the degree value d_i of the node i th. We define the *extended Laplacian* matrix of elements $\tilde{l}_{ij}$ as $\tilde{L} = L + I$, with I being the identity matrix of opportune dimensions. The matrix $\tilde{L}$ associated to the undirected graph is real and symmetric with real spectrum $\lambda(\tilde{L})$.

In the recent years various dynamic models have been used by a number of researchers to model a wide range of queueing and contention systems. Several variants of the fluid model have been extensively used for network performance

evaluation and control (e.g., [2, 3, 6, 13]). Herein, the main objective is to consider a low order complexity model of multi-bottleneck capturing the essential dynamics of network behavior which is suitable for a distributed consensus-based rate control design. Moreover, we would consider in the model the presence of time delays in the sources data flow. A time delay is due the time elapsed between a rate command signal by a switch controller and the actual time this rate is set. This delay from the control input to the regulated output is the sum of two delays (backward delay τ_b from controller to source and forward delay τ_f from source to controller) named the *Round-Trip Time delay* RTT. Considered a network graph consisting by a set of congested nodes $N = \{1, 2, \dots, n\}$ and $M = \{1, 2, \dots, m\}$ accessing source classes by a specific source–destination path, the source-link interconnections can be described by the routing matrix

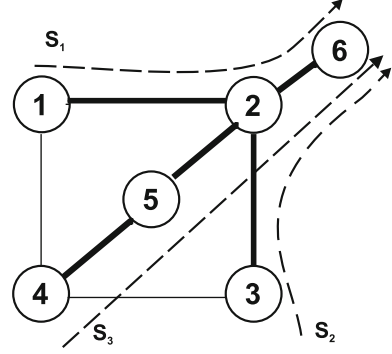
$$R_{ij}(s) = \begin{cases} e^{-s\tau_{i,j}}, & \text{if source } j \text{ traverses link } i; \\ 0, & \text{otherwise.} \end{cases}$$

with $\tau_{i,j}$ denoting the delay of the source j with respect to (w.r.t) link i , and s representing the Laplace complex variable. Notice that as the congestion at each router is just due to the output link congestion, herein we will use indifferently the terms congested node and congested link. In what follows, for sake of brevity, we use the term source to briefly denote source class, namely, the set of sources characterized by the same service class or priority. We introduce the forward routing matrix $R_{ij}^f(s)$ of elements $e^{-s\tau_{fi,j}}$ with $\tau_{fi,j}$ is the forward time delay from source j to link i , and the backward routing matrix $R_{ij}^b(s)$ of elements $e^{-s\tau_{bi,j}}$ with $\tau_{bi,j}$ is the backward time delay from link i to source j . In this way, the source j th has w.r.t link i th the round-trip time: $RTT_{i,j} = \tau_{fi,j} + \tau_{bi,j}$. Starting from the simple fluid queue model of a single bottleneck and multiple time delayed sources widely used in the literature (i.e., [2–4, 6, 13]), and denoted $q_i(t)$ the queue length at the bottleneck link i th and $r_{i,j}(t)$ being the nonnegative data flow rate of the j th source accessing to the i th bottleneck link, the network open loop dynamic model is described by

$$\dot{q}_i(t) = \sum_{j \in \bar{S}_i} r_{i,j}(t - \tau_{fi,j}) - c_i(t), \quad (2.1)$$

for $i \in N$, $j \in \bar{S}_i = \{v \in M : v \text{ traverses the link } i\text{-th}\}$ and $c_i(t)$ being the rate at which data is sent out from the link i th. In order to recast the congestion problem in terms of cooperative control concept, we introduce the set of *virtually bottleneck neighbors* of i th link defined as $N_i = \{k \in N : \bar{S}_i \cap \bar{S}_k \neq \emptyset, a_{ik} = 1\}$. In other words, *virtually bottleneck neighbors* are bottlenecks sharing source paths. For instance referring to Fig. 2.2, links $1 \mapsto 2$ and $2 \mapsto 6$ share the path of the source S_1 and so they are *virtually bottleneck neighbors*. In the same way the links $4 \mapsto 5$, $5 \mapsto 2$ and $2 \mapsto 6$ are one-step virtually neighbors. On the other side, links $1 \mapsto 4$ and $4 \mapsto 3$, although are physical connected, they are not virtually bottleneck neighbors as do not share any source path. The overall graph composed of bottleneck

Fig. 2.2 Network graph.
The overlay virtual graph is
denoted by highlighted *solid*
lines



nodes and their virtual neighbors is defined as the *overlay virtual graph* (denoted by highlighted solid lines in Fig. 2.2).

According to the rate control strategies presented in the literature (see i.e., [7, 8] and references therein), we consider that source rate $r_{i,j}(t)$ will be assigned to the source j th by a feedback controller $u_{i,j}$ located at the bottleneck i th. Therefore starting from the open loop model (2.1), we get the following closed-loop model:

$$\dot{q}_i(t) = \sum_{j \in \tilde{S}_i} u_{i,j}(t - RTT_{i,j}) - c_i(t), \quad (2.2)$$

for $i = 1, \dots, n$ and with $RTT_{i,j}$ is the round-trip time of the source j th w.r.t link i th. We note that the source rate commands $u_{i,j}$ should satisfy the constrain on the aggregate available rate u_i computed by the controller. So if $u_{i,j} = k_{i,j}u_i$, $k_{i,j}$ are nonnegative controller gains to be designed so that $\sum_{j \in \tilde{S}_i} k_{i,j} \leq 1$. Let us assume that the final allocated rate r_j to the source j th, $\forall j \in M$, is equal to the minimum rate value u_{mj} among the rate values assigned by the links along the path of its flow (i.e., $r_j = u_{mj} = \min_i u_{i,j}$ with $i \in B_j = \{l \in N : l \text{ is a bottleneck for the source } j\}$). Because the minimum operation is taken over a finite number of links and each flow j has at least one bottleneck on its path, there should exist u_{mj} , $\forall j \in M$. Therefore, the vector $r = [r_1, \dots, r_m]^T$ denotes the allocated sources rates.

We do the following assumptions:

Assumption 2.1 The sources are persistent until the closed-loop system reaches steady state meaning that the source always has enough data to transmit at the allocated rate.

Assumption 2.2 All links are bottleneck for at least one source $j \in M$. Specifically the link i th is a bottleneck for a given source j th if and only if the source j th has the maximum rate among all sources using the link i th (i.e., $r_j \geq r_{j'}$, for all $j' \in \tilde{S}_i$). In other words, the bottleneck i th is a link which is limiting for a given allocation r_j of the source j th. Moreover, being all links bottleneck for at least one source we can assume $c_i(t) = c_i$ for all i , with c_i to be the i th link capacity (i.e., maximum link rate).

2.3 Consensus-Based Cooperative Rate Control Scheme: Stability and Convergence Results

Herein we detail the consensus-based cooperative rate control scheme (CRC). The controller is implemented at the bottleneck node and adjusts sources rate according to both its own congestion level (i.e., queue length) and that of its *virtually bottleneck neighbors*. We state the first stability result.

Theorem 2.1 *Consider a n -links m -sources communication network described by (2.2). Chosen the consensus-based cooperative control action*

$$u_{i,j}(t) = k_{i,j} \sum_{k \in N_i \cup \{q_0\}} (q_k(t) - q_i(t)) + k_{f,i,j} \tilde{c}_i(t), \quad (2.3)$$

then it results:

(a) *the network is globally asymptotically stable if*

$$k_{i,j} \leq \frac{\pi}{2 \mid \bar{S}_i \mid RTT_{M_i} \lambda_M}, \quad (2.4)$$

$\forall i \in N, \forall j \in \bar{S}_i$, with $RTT_{M_i} = \max\{RTT_{i,j}, j \in \bar{S}_i\}$, λ_M maximum eigenvalue of $\tilde{L}$, $k_{f,i,j} \tilde{c}_i(t)$ is a feedforward action for link capacity allocation with gain $k_{f,i,j}$ and $\tilde{c}_i$ is an estimation of the link capacity;

(b) *the network queues asymptotically converge to the same set point value q_0 (i.e. network queues balancing).*

The control law $u_{i,j}(t)$ is composed of the feedback cooperative term $u_{i,j}^{fb}(t) = k_{i,j} \sum_{k \in N_i \cup \{q_0\}} (q_k(t) - q_i(t))$ (including the setpoint term in q_0) and of the feedforward action term $u_{i,j}^{ff}(t) = k_{f,i,j} \tilde{c}_i(t)$. Hence, we have to design feedback gains $k_{i,j}$ and feedforward gains $k_{f,i,j}$.

Proof (a) Notice that $\tilde{c}_i$ is the link capacity estimation that in general it is easily carried out at the node by local measurements. From Assumption 2.2 it results $\tilde{c}_i(t) = c_i$ and thus $u_{i,j}^{ff}(t) = k_{f,i,j} c_i(t) = k_{f,i,j} c_i$. We design $u_{i,j}^{ff}(t)$ in order to allocate the i th link capacity c_i , fulfilling the constraint that the total capacity made available to sources is less or equal than c_i . In particular, choosing $k_{f,i,j}$ according to

$$k_{f,i,j} = \frac{w_j}{\sum_{k \in \bar{S}_i} w_k} \quad (2.5)$$

with w_j being the priority weight associated to the source class j th, the amount of capacity allocated to the j th source then results: $u_{i,j}^{ff} = \frac{w_j}{\sum_{k \in \bar{S}_i} w_k} c_i$. In this way, the allocation of the available capacity among sources guarantees not only that the allocated capacity is within bounds but also that the allocation is proportionally fair. With proportional fairness, sources with greater weights w_j are allocated a larger

amount of capacity. We can interpret w_j as a preassigned level of Quality of Service within the ordinary class to the source j th. Thus (2.5) can be used for feedforward gains $k_{f_{i,j}}$ design purpose in order to fair allocate the available capacity c_i on the base of source priorities or differentiated service requirements. This easily allows to apply the approach to DiffServ network. In the case of equal w_j for all j , the resulting capacity allocation is *max-min fair* with all sources getting the same resource quota. In what follows we present a simple static fair allocation strategy used by sink for avoiding congestion and guaranteeing fair capacity allocation from sensor nodes.

Let a link i th with capacity c_i , the simplest method fulfilling the constraint that the total bandwidth made available to sensors traversing i is less or equal than c_i , consists of allocating an amount of capacity to the j -th sensor, $r_{i,j}$, so that:

$$\frac{w_j}{\sum_{k \in \bar{S}_i} w_k} = \frac{r_{i,j}}{c_i}$$

i.e., setting

$$r_{i,j} = \frac{w_j}{\sum_{k \in \bar{S}_i} w_k} c_i \quad (2.6)$$

with w_j is the weight associated to the source or sensor j th. Following the approach shown in [31], for a given number of $\bar{S}_i$ sensors and a fixed set of weights $w = (w_1, \dots, w_{\bar{S}_i})^T$, a given allocation of resource is proportionally fair if it solves the following optimal problem:

$$\max_{r_{i,j}} \sum_{j \in \bar{S}_i} w_j \log r_{i,j} \quad (2.7)$$

subject to the constraint

$$\sum_{j \in \bar{S}_i} r_{i,j} \leq c_i \quad (2.8)$$

over $r_{i,j} \geq 0$. As observed in [31], if we assume that the user utility scales as logarithm of the allocated capacity, then (2.7) corresponds to the maximization of the collective log utility. Moreover, constraint (2.8) guarantees the use of all available capacity, and therefore good link utilization.

The Lagrangian for the problem can be expressed as

$$L(r_{i,j}, \lambda) = \sum_{j \in \bar{S}_i} w_j \log r_{i,j} + \lambda \left(c_i - \sum_{j \in \bar{S}_i} r_{i,j} \right)$$

with λ is the Lagrange multiplier. Then for all $j \in \bar{S}_i$,

$$\frac{\partial L}{\partial r_{i,j}} = \frac{w_j}{r_{i,j}} - \lambda$$

and considering the constraint on the capacity, the unique optimum point can be derived to be

$$r_{i,j}^o = \frac{w_j}{\lambda} \quad (2.9)$$

with

$$\lambda = \frac{\sum_{k \in \bar{S}_i} w_k}{c_i}.$$

Therefore, the allocation of the available resource among users according to (2.9) guarantees not only that the allocated capacity is within bounds but also that the allocation is proportionally fair according to the criterion given in [31]. With proportional fairness, users with greater weights w_j are allocated a larger amount of capacity, causing a heavy reduction in the allocation for other users. We can interpret the components of w as prespecified sensor priority requirements on the Quality of Service (e.g., fault detection signaling, vision monitoring/control). Thus, $u_{i,j}^{ff}$ can be used by the resource manager to govern the relative fair allocation of capacity among users based on their priorities by designing the feedforward gains $k_{f_{i,j}} = \frac{w_j}{\sum_{k \in \bar{S}_i} w_k}$. In particular, in the case of equal w components, the resulting resource allocation is max-min fair. Without loss of generality, we will assume $k_{i,j} = k_i \forall j \in \bar{S}_i$ and $i \in N$. In so doing all sources sharing a common link receive the same rate quota of the feedback term $u_{i,j}^{fb}(t)$. This means that there is no a particular differentiated services for the sources accessing to the link i th. Substituting (2.3) in the closed-loop equation (2.2) and being $\sum_{j \in \bar{S}_i} k_{f_{i,j}} c_i(t) = \sum_{j \in \bar{S}_i} k_{f_{i,j}} c_i = c_i$ (having chosen $k_{f_{i,j}}$ according to (2.5)), then it results:

$$\begin{aligned} \dot{q}_i(t) &= \sum_{j \in \bar{S}_i} k_{i,j} \sum_{k \in N_i \cup \{q_0\}} (q_k(t - RTT_{i,j}) - q_i(t - RTT_{i,j})) \\ &= \sum_{j \in \bar{S}_i} k_i \left(\sum_{k \in N_i} (q_k(t - RTT_{i,j}) - q_i(t - RTT_{i,j})) + q_0 - q_i(t - RTT_{i,j}) \right) \\ &= k_i \sum_{j \in \bar{S}_i} \left(\sum_{k \in N_i} (q_k(t - RTT_{i,j}) - q_i(t - RTT_{i,j})) - q_i(t - RTT_{i,j}) \right) + k_i | \bar{S}_i | q_0. \end{aligned} \quad (2.10)$$

Notice that $\sum_{k \in N_i} (q_k(t) - q_i(t)) - q_i(t)$ represents the i th element of product $-\tilde{L}q(t)$ with $q(t) = [q_1, \dots, q_n]^T$ being the vector of network queue lengths at the time t . Let $k = [k_1, k_2, \dots, k_n]^T$ and defined $\tilde{R}(s)$ the delay diagonal matrix with $\sum_{j \in \bar{S}_i} e^{-sRTT_{i,j}}$ on the i th diagonal position, $K = \text{diaggen}\{k\}$ the controller

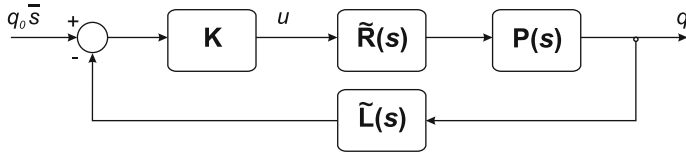


Fig. 2.3 Block diagram of CRC system

feedback gain matrix, $P(s) = \text{diaggen}\{\frac{1_n}{s}\}$ the queue process, then the controlled network reduces to the feedback control system in Fig. 2.3 with q_0 being the reference queue length and $\bar{s} = [|\bar{S}_1|, \dots, |\bar{S}_n|]^T$ is the vector of the number of the sources traversing the links. The resulting return ratio transfer function is: $H(s) = K \tilde{R}(s) P(s) \tilde{L}$. Moreover it results: $\sigma(H(s)) \subset f(K \tilde{R} P \tilde{L}) \subset f(K \tilde{R} P)$ $f(\tilde{L}) = Co\{k_i \tilde{r}_i p_i\} Co\{\lambda(\tilde{L})\} \subseteq Co\{k_i \sum_{j \in \bar{S}_i} \frac{e^{-s R T T_{i,j}}}{s}\} [\lambda_m, \lambda_M]$.

Indeed, being the matrices *normal* [32], the first and the second above inclusions follow from the spectral containment and field values properties while the next equality follows from the normality property [32]. We note that the real part of the set $Co\{k_i \sum_{j \in \bar{S}_i} \frac{e^{-s R T T_{i,j}}}{s}\}$ is lower limited by the point $-k_i R T T_{M_i} | \bar{S}_i | \frac{2}{\pi}$ setting $R T T_{i,j} = R T T_{M_i}$, $\forall j \in \bar{S}_i$ and $s \cdot R T T_{M_i} = j \frac{\pi}{2}$.

Hence, chosen $k_{f_{i,j}}$ according to (2.5), if (2.4) holds then $H(s)$ do not intersect $(-\infty, -1]$ for all s and from the Generalized Nyquist criterion [33] the closed-loop system is global asymptotically stable. This completes the proof (a).

Therefore, the CRC control action $u_{i,j}$ at the link i th allocates the link capacity (by the feedforward action) and regulates the sources rate according its level of congestion q_i and the level of congestion q_k , $k \in N_i$ as depicted in Fig. 2.4, in the spirit of the cooperation approach between virtually bottleneck neighbors.

Proof (b) One expectant goal of the proposed consensus-based rate control law is to bring the network to the balanced desired equilibrium such that $q_i = q_0$, for all $i \in N$, with q_0 being the target queue value. Indeed, the network under the proposed consensus-based rate control law (2.3) presents the equilibrium point $\bar{q}_i = q_0$, for all $i \in N$, as easily results from the closed-loop model equation (2.2). We will show the convergence of the network to the above equilibrium point by computing the set point error $e(t)$ between the queue values and the step reference $\frac{q_0}{s} 1_n$. Let $S_o(s)$ the sensitivity function of the closed-loop system, from the final value theorem it results:

$$\begin{aligned} \lim_{t \rightarrow \infty} e(t) &= \lim_{s \rightarrow 0} s E(s) = \\ &= \lim_{s \rightarrow 0} s S_o(s) \frac{q_0}{s} 1_n = \lim_{s \rightarrow 0} s (sI + K \tilde{R}(s) \tilde{L})^{-1} q_0 1_n = 0. \end{aligned}$$

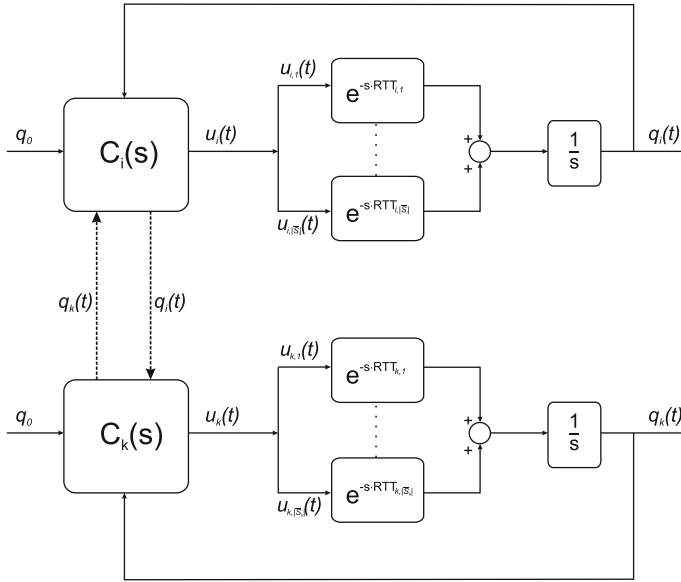


Fig. 2.4 Block diagram of cooperating i th and k th bottleneck nodes

The convergence to zero of $e(t)$ follows considering that K is a stabilizing controller from the **proof** (a), $S_o(0) = 0$ and that $K\tilde{R}(0)\tilde{L}$ is an invertible matrix.¹ This completes the proof of Theorem 2.1. ■

Therefore, the CRC control action $u_{i,j}$ at the link i th allocates the link capacity (by the feedforward action) and regulates the sources rate according its level of congestion q_i and the level of congestion q_k , $k \in N_i$ as depicted in Fig. 2.4, in the spirit of the cooperation approach between virtually bottleneck neighbors.

We note that switches need to exchange only queue level with its virtual neighbors for implementing cooperative control. We note that pinning cooperative control operates locally at each bottleneck node. In particular the one-step bottleneck neighbors inform one with each other of own queue level and set the respective controlled sources rates according the closed-loop scheme of Fig. 2.3.

The proposed control scheme guarantees queue balancing and set point regulation by opportunely tuning the feedback and feedforward gains (i.e., $k_{f,i,j} = \frac{w_j}{\sum_{k \in \tilde{S}_i} w_k}$ and $k_{i,j} < \frac{\pi}{2|\tilde{S}_i|RT_{M_i}\lambda_M}$) on the base of network and source features (i.e., virtual graph topology, source priority, maximum round-trip time).

The feedback design condition (2.4) depends on the round trip time and network topology information that can be no easily available by the designer. In what follows we introduce corollaries of Theorem 2.1 in order to give practical control design law depending on more accessible network parameters. This is particularly appealing

¹Indeed it is strictly diagonally dominant matrix and hence for Geršgorin theorem [32] is invertible.

for the NMCSs based on wired technology because the time delays may be easily estimated.

Corollary 2.1 *Consider a n -links m -sources communication network described by (2.2). Let $\tau_{p_{m_{ij}}}$ the propagation delay of the connection between the source j th and the link i th and chosen the cooperative control action (2.3), then the network is globally asymptotically stable if*

$$k_{i,j} < \frac{\pi}{2 \mid \bar{S}_i \mid (2\tau_{p_{m_i}} + b_i/c_i)(2\bar{S}_M + 1)}, \quad (2.11)$$

$\forall i \in N, \forall j \in \bar{S}_i$, where b_i is the buffer size, $\tau_{p_{m_i}} = \max\{\tau_{p_{m_{ij}}}, j \in \bar{S}_i\}$, $\bar{S}_M = \max_i \mid \bar{S}_i \mid$. Moreover, the network queues asymptotically converge to the same set point value q_0 (i.e. network queues balancing).

Proof Let b_i the buffer size of the i th link, thus $RTT_{M_i} = 2\tau_{p_{m_i}} + b_i/c_i$ with $\tau_{p_{m_i}}$ is the maximum propagation delay among the sources accessing to the i th link. Moreover, from the Gersgorin's theorem all the eigenvalues of $\tilde{L}$ are located in the union of the following n disks: $\tilde{L}_i = \{z \in \mathbb{C}, \|z - \tilde{l}_{i,i}\| \leq \sum_{j=1, j \neq i}^n \|\tilde{l}_{i,j}\|\} = \{z \in \mathbb{C}, \|z - d_i - 1\| \leq \sum_{j=1, j \neq i}^n \|a_{i,j}\|\}$, $i = 1, \dots, n$. Let d_M be the maximum degree of the virtual graph, and defined the largest disk radius $r_M = \max_i \sum_{j=1, j \neq i}^n \mid a_{i,j} \mid = \max_i d_i = d_M$, being $\tilde{L}$ real and symmetric matrix then follows: $\forall \lambda \in \lambda(\tilde{L}), 1 \leq \lambda \leq 2d_M + 1$. Let $\bar{S}_M = \max_i \mid \bar{S}_i \mid$ and being $d_M = \bar{S}_M$ then $\lambda_M \leq 2\bar{S}_M + 1$ and hence we can recast (2.4) into the relation (2.11). The proof of the second part of the Corollary follows the same arguments given in the Theorem 2.1. ■

Corollary 2.2 *Consider a n -links m -sources communication network described by (2.2). Let $\tau_{p_{m_{ij}}}$ the propagation delay of the connection between the source j th and the link i th and chosen the cooperative control action (2.3), then the network is globally asymptotically stable if*

$$k_{i,j} < \frac{\pi}{2 \mid \bar{S}_i \mid (2\tau_{p_{m_i}} + b_i/c_i)\hat{S}_M}, \quad (2.12)$$

$\forall i \in N, \forall j \in \bar{S}_i$, where b_i is the buffer size, $\tau_{p_{m_i}} = \max\{\tau_{i,j}, j \in \bar{S}_i\}$, $\hat{S}_M = \max\{\mid S_i \mid + \mid S_j \mid : (i, j) \in N \times N\}$. Moreover, the network queues asymptotically converge to the same set point value q_0 (i.e. network queues balancing).

Proof Let λ_M the maximum eigenvalue of the extended Laplacian $\tilde{L}$, if the virtual network graph associated to $\tilde{L}$ is connected and bipartite then results [34]: $\lambda_M \leq \max\{d_i + d_j : (i, j) \in E \times E\}$. Because the degree of each node i th of the virtual graph corresponds to the number of the sources passing through the link i th (i.e., $d_i = \mid S_i \mid$) then the result follows. ■

The above Corollaries can be used for distributed network design because they relate network parameters (e.g., link capacity c_i , buffer size b_i , propagation delay $\tau_{p_{m_i}}$ and

number of source classes) to feedback controller gains $k_{i,j}$. Notice that usually the number of sources is much larger than the number of links ($m \gg n$). This implies that the virtual graph is undirected and connected or at least is composed of the number of connected and undirected clusters. Also the assumption of bipartite graph is a realistic assumption for a virtual graph associated to the communication network. Indeed, it has shown as communication physical and overlay network topologies have scale free property (see i.e., [35] and references therein) and how those networks can be reviewed as bipartite graph [36].

2.4 Performance Issues

In what follows we analyze the CRC performance in terms of link utilization, set point regulation and fairness of the proposed controller.

2.4.1 Set Point Regulation, Queue Balancing, and Link Utilization

- *Set point regulation and queue balancing*

From the proof of part (b) of Theorem 2.1 we have shown that the CRC algorithm assures queues set point regulation to a desired $0 < q_0 < \min_i b_i$ with resulting network queue balancing and set point regulation. Moreover this avoids packet dropping for buffer overflow.

- *Link utilization*

From Theorem 2.1, if the cooperative control law (2.3) is applied, at the steady state results: $\sum_{j \in \tilde{S}_i} r_j = c_i$ implying that the capacity at the link i th is fully utilized.

2.4.2 Fairness

In a shared environment the throughput for a source depends upon the demands by other sources. The most commonly criterion for the correct share of bandwidth for sources in network environment is the so called max-min allocation [37]. It provides the maximum possible bandwidth to the source receiving the least among all contending sources. Notice that max-min allocation is both fair and efficient in the sense that all sources get an equal share on every link and that each link is utilized to the maximum load possible. In what follows we will show as CRC achieves max-min fair resource allocation.

Definition 2.1 A vector of allocated rates $r = [r_1, \dots, r_m]^T$ is feasible if $r_j \geq 0, \forall j \in M$ and $\sum_{j \in \tilde{S}_i} r_j \leq c_i$, for all $i = 1, \dots, n$.

Definition 2.2 ([38]) A vector r is max–min fair if and only if it is feasible, and for each $j \in M$ and for any other feasible vector allocation rate $\bar{r}$ for which $r_j < \bar{r}_j$, there is some j' such that $r_j \geq r_{j'} > \bar{r}_{j'}$.

In other term, a max–min fair rate vector is such that for every rate r_j , any attempt to increase r_j must result in a decrease of another rate $r_{j'}$, for which $r_j \geq r_{j'}$ in order to maintain feasibility. In this way it is given priority to flows with small rate values.

Proposition 2.1 *The vector r of the source rates allocated by the consensus-based control (2.3) is max–min fair.*

Proof We introduce the set of flows bottlenecked at the link $\hat{i}$ th, $\bar{S}_{\hat{i}_b} = \{j \in M : r_j = u_{mj} = u_{\hat{i},j}\}$, and the set of all flows not bottlenecked at link $\hat{i}$ th and traversing it $S_{\hat{i}_{nb}} = \{j \in M : r_{i,j} = u_{\bar{i},j} \text{ and } u_{\bar{i},j} < u_{\hat{i},j}\}$ with $\bar{i} \in B_j$ is some bottleneck ($\neq \hat{i}$) for the flow j th. At steady state the allocate rates vector r is feasible since feasibility is a necessary condition for stability and being the cooperative control (2.3) a stabilizing control for the network (2.2) from Theorem 2.1. We assume that we can increase the rate value of flow $j \in \bar{S}_{\hat{i}_b}$ which is bottlenecked at the link $\hat{i}$ th and therefore it exists at least another flow j' sharing with the flow j th the link $\hat{i}$ th. For all $j' \in \bar{S}_{\hat{i}}$, results $r_{j'} \leq r_j$ being $r_{j'} < r_j$ for $j' \in \bar{S}_{\hat{i}_{nb}}$ and $j' \neq j$ or $r_{j'} = r_j$ for $j' \in \bar{S}_{\hat{i}_b}$ and $j' \neq j$. Using the notation just introduced, if the Assumption 2.2 holds and if the cooperative control law (2.3) is applied thus at steady-state results $\sum_{j \in \bar{S}_{\hat{i}_b}} r_j + \sum_{j \in \bar{S}_{\hat{i}_{nb}}} r_j = c_{\hat{i}}$.

Therefore, for every rate r_j bottlenecked at the link $\hat{i}$ th, any attempt to increase r_j must result in a decrease of another rate $r_{j'}$ for which $r_j \geq r_{j'}$ in order to maintain feasibility. Finally according to Definition 2.2, the steady-state rate vector is max–min fair. ■

Remark 2.1 As stated above we have shown the max–min property of the allocated rate by the proposed controller. We have considered max–min fair allocation as it is usually considered the common way to allocate the same quota capacity to all sources with the same priority. Anyway by using Eq. (2.5), the designer can set the feedforward gains in order to get a proportional fair capacity allocation so that each source j has an amount of allocated capacity proportionally to its weight w_j . The weight w_j can be seen as a preassigned level of Quality of Service (set by the network manager or bought by the end-user).

2.5 Implementation Issues

The proposed consensus-based control law (2.3) can be implemented by end-to-end or hop-by-hop mechanism by using existing protocols of wireless or wired networks. Specifically the information about the queue length can be exchanged among nodes (Fig. 2.4) by using the available field of the specific control packet in a given protocol: for example (1) over ATM protocol, the RM cell can be used; (2) over TCP

protocol, it is possible to use the same packet signaling proposed in [9]; (3) over wireless networks, it is possible to use the “HELLO” packet, that is a special packet periodically sent from a node to discover neighboring routers; (4) over Content Delivery Networks (CDNs), a control signal at the application layer can be adopted. The use of end-to-end and hop-by-hop mechanism for implementing consensus-based algorithms at the network layer will be presented in Chap. 4.

References

1. Imer, O.C., Compans, S., Basar, T., Srikant, R.: Available bit rate congestion control in ATM networks: developing explicit rate control algorithms. *IEEE Control Syst. Mag.* **21**, 38 (2001)
2. Mascolo, S.: Smith’s principle for congestion control in high-speed data networks. *IEEE Trans. Autom. Control* **45**, 358–364 (2000)
3. Quet, P.H., Ataslar, B., Iftar, A., Ozbay, H., Kang, T., Kalyanaraman, S.: Rate based flow controllers for communication networks in the presence of uncertain time varying multiple time delays. *Automatica* **38**, 917–928 (2002)
4. Sichiitiu, M.L., Bauer, P.H.: Asymptotic stability of congestion control systems with multiple sources. *IEEE Trans. Autom. Control* **51**, 292–298 (2006)
5. Iwatani, J., Yakoh, T.: A detection method of active queue management in communication paths. *IEEE Int. Conf. Ind. Technol. (ICIT)* (2010)
6. Holot, C.V., Misra, V., Towsley, D., Gong, W.: Analysis and design of controllers for AQM routers supporting TCP flows. *IEEE Trans. Autom. Control* **47**, 945–959 (2002)
7. Balakrishnan, H., Dukkupati, N., McKeown, N., Tomlin, C.J.: Stability analysis of explicit congestion control protocols. *IEEE Commun. Lett.* **11**, 823–825 (2007)
8. Kelly, F., Raina, G., Voice, T.: Stability and fairness of explicit congestion control with small buffers. *ACM SIGCOMM Comput. Commun. Rev.* (2008)
9. Dukkupati, N., Kobayashi, M., Zhang-Shen, R., McKeown, N.: Processor sharing flows in the internet. In: *Thirteenth International Workshop on Quality of Service* (2005)
10. Paek, J., Govindan, R.: RCRT: rate-controlled reliable transport for wireless sensor networks. *Sensys.* (2007)
11. Akan, O., Akyildiz, I.F.: Event-to-sink reliable transport in wireless sensor networks. *IEEE/ACM Trans. Netw.* **13**, 1003–1016 (2005)
12. Koutsopoulos, I., Tassiulas, L.: Optimal transmission rate control policies in a wireless link under partial state information. *IEEE Trans. Autom. Control* **55**, 127–131 (2010)
13. Mastrocristino, T., Tesoriere, G., Grieco, L.A., Boggia, G., Palattella, M.R., Camarda, P.: Congestion control based on data-aggregation for wireless sensor networks. In: *IEEE International Symposium on Industrial Electronics* (2010)
14. Wirz, R., Marin, R., Ferre, M., Barrio, J., Claver, J.M., Ortego, J.: Bidirectional transport protocol for teleoperated rob. *IEEE Trans. Ind. Electron.* **56**, 3772–3781 (2009)
15. Yashiro, D., Yakoh, T., Ohnishi, K.: End-to-end flow control using PI controller for force control system over TCP/IP network. In: *7th IEEE International Conference on Industrial Informatics* (2009)
16. Lu, Q., Han, Q.-L.: Cooperative control of a multi-robot system for odor source localization. *IECON 2011 - 37th Annual Conference on IEEE Industrial Electronics Society* (2011)
17. Andaluz, V., Rampinelli, V.T.L., Roberti, F., Carelli, R.: Coordinated cooperative control of mobile manipulators. *IEEE International Conference on Industrial Technology* (2011)
18. Rezaee, H., Abdollahi, F.: Mobile robots cooperative control and obstacle avoidance using potential field. *IEEE/ASME International Conference on Advanced Intelligent Mechatronics* (2011)

19. Hu, S., Wang, H., Kwong, S., Kuo, C.J.: Novel rate-quantization model-based rate control with adaptive initialization for spatial scalable video coding. *IEEE Trans. Ind. Electron.* **59**, 1673–1684 (2012)
20. Corradini, M.L., Fossi, V., Giantomassi, A., Ippoliti, G., Longhi, S., Orlando, G.: Minimal resource allocating networks for discrete time sliding mode control of robotic manipulators. *IEEE Trans. Ind. Inf.* **8**, 733–745 (2012)
21. Chakareski, J.: In-network packet scheduling and rate allocation: a content delivery perspective. *IEEE Trans. Multimed.* **13**, 1092–1102 (2011)
22. Collotta, M., Bello, L.L., Toscano, E., Mirabella, O.: Dynamic load balancing techniques for flexible wireless industrial networks. In: *IECON 2010 - 36th Annual Conference on IEEE Industrial Electronics Society* (2010)
23. Murray, R.M.: Recent research in cooperative control of multivehicle systems. *J. Dyn. Syst. Meas. Control* **129** (2007)
24. Zhang, H., Lewis, F.L., Qu, Z.: Lyapunov, adaptive, and optimal design techniques for cooperative systems on directed communication graphs. *IEEE Trans. Ind. Electron.* **59**, 3026–3041 (2012)
25. Cheng, T.M., Savkin, A.V.: Decentralized control of mobile sensor networks for asymptotically optimal blanket coverage between two boundaries. *IEEE Trans. Ind. Electron.* **9**, 365–376 (2013)
26. Zhan, J., Li, X.: Flocking of multi-agent systems via model predictive control based on position-only measurements. *IEEE Trans. Ind. Inf.* **9**, 377–385 (2013)
27. Takahashi, R., Kinoshita, H., Murata, T., Tamura, J., Sugimasa, M., Komura, A., Futami, M., Ichinose, M., Ide, K.: Output power smoothing and hydrogen production by using variable speed wind generators. *IEEE Trans. Ind. Electron.* **57**, 485–493 (2010)
28. Panagi, P., Polycarpou, M.M.: A coordinated communication scheme for distributed fault tolerant control. *IEEE Trans. Ind. Inf.* **9**, 386–393 (2013)
29. Zhang, N., Yang, M., Jing, Y., Zhang, S.: Congestion control for diffserf network using second-order sliding mode control. *IEEE Trans. Ind. Electron.* **56**, 3330–3336 (2009)
30. Das, A.K., Fierro, R., Kumar, R.V., Ostrowski, J.P., Spletzer, J.: A vision-based formation control framework. *IEEE Trans. Robot. Autom.* **18**, 813–825 (2002)
31. Kelly, F., Maulloo, A.K., Tan, D.K.H.: Rate control for communication networks: shadow prices, proportional fairness and stability. *J. Oper. Res. Soc.* **74**, 237–252 (1998)
32. Horn R.A., Johnson, C.R.: *Topics in Matrix Analysis*. Cambridge University Press, Cambridge (1995)
33. Desoer, C.A., Yang, Y.T.: On the generalized Nyquist stability criterion. *IEEE Trans. Autom. Control* **25**, 187–196 (1980)
34. Mohar, B.: The Laplacian spectrum of graphs. *Graph theory, combinatorics, and applications. Graph Theory Comb. Appl.* **2** (1991)
35. Siganos, G., Faloutsos, M., Faloutsos, P., Faloutsos, C.: Power-laws and the AS-level internet topology. *IEEE/ACM Trans. Netw.* **11**, 514–524 (2003)
36. Ohkubo, J., Tanaka, K., Horiguchi, T.: Generation of complex bipartite graphs by using a preferential rewiring process. *Phys. Rev. E Stat. Nonlinear Soft Matter Phys.* (2005)
37. Jain, R.: *The Art of Computer Systems Performance Analysis*. Wiley, New York (1991)
38. Bertsekas, D., Gallager, R.: *Data Networks*. Prentice-Hall, Englewood Cliffs (1992)

Multilayer Control of Networked Cyber-Physical Systems
Application to Monitoring, Autonomous and Robot
Systems

Manfredi, S.

2017, XVIII, 137 p. 75 illus., 57 illus. in color., Hardcover

ISBN: 978-3-319-41645-8