A Novel Congestion Control Scheme in Network-on-Chip Based on Best Effort Delay-Sum Optimization

Fahimeh Jafari[1,2], Mohammad S. Talebi[2], Ahmad Khonsari[3,2], Mohammad H. Yaghmaee[1]

[1]*Ferdowsi University of Mashhad, Mashhad, Iran.,*
[2]*IPM, School of Computer, Tehran, Iran.,*
[3]*ECE Department, University of Tehran, Tehran, Iran.,*
jafari@ipm.ir, mstalebi@ipm.ir, ak@ipm.ir, hyaghmae@ferdowsi.um.ac.ir

Abstract

With the advances of the semiconductor technology, the enormous number of transistors available on a single chip allows designers to integrate dozens of IP blocks together with large amounts of embedded memory. This has been led to the concept of Network on a Chip (NoC), in which different modules would be connected by a simple network of shared links and routers and is considered as a solution to replace traditional bus-based architectures to address the global communication challenges in nanoscale technologies. In NoC architectures, controlling congestion of the best effort traffic will continue to be an important design goal. Towards this, employing end-to-end congestion control is becoming more imminent in the design process of NoCs. In this paper, we introduce a centralized algorithm based on the delay minimization of Best Effort sources. The proposed algorithm can be used as a mechanism to control the flow of Best Effort source rates by which the sum of propagation delays of network is to be minimized.

1. Introduction

With the emergence of complex VLSI chips, the designers are facing several new challenges. Nowadays, *application-specific integrated circuits* (ASICs) have evolved into *systems-on-chip* (SoCs), where dozens of predesigned IP cores are assembled together to form large chips with complex functionality.

A recently proposed platform for the on-chip interconnects is the *network-on-chip* (NoC) architecture, where the IPs are usually placed on a grid of tiles and networking protocols governs the communication between tiles. Such a regular structures are very attractive because they can offer well-controlled electrical parameters, which enable high-performance circuits by reducing the latency and increasing the bandwidth. In fact, NoCs provide enhanced performance and scalability, in comparison with previous communication architectures. The advantages of NoC are achieved thanks to efficient sharing of wires and a high level of parallelism [1].

The provision of *Quality-of-Service* (QoS) in NoC's environment is currently the focus of much discussion in research community. NoCs are expected to serve as multimedia servers and are required not only to carry Best Effort (BE) traffic, but also Guaranteed Service (GS) traffic which requires tight performance constraints such as necessary bandwidth and maximum delay boundaries. Networks with BE services must choose a mechanism to avoid congestion. Congestion control in NoCs is a novel issue and usually studied regarding minimizing the network cost (in delay, area and power) or maximizing network utility while maintaining the required QoS, as we will focus on it in more detail later.

2. Related Works

During the past few years, many strategies for congestion control have been proposed for off-chip networks [2-5]. Congestion control for on-chip networks is still a novel issue, however this problem has been investigated by several researchers [6]-[8]. In [6], a prediction-based congestion control strategy for on-chip networks has been proposed where each router predicts buffer occupancies to detect future congestion problems. In [7] the link utilization has been used as congestion measure and the controller determines the appropriate loads for the BE sources. Dyad [8] overcomes the congestion by switching from

978-0-7695-3125-0/08 $25.00 © 2008 IEEE
DOI 10.1109/I-SPAN.2008.45

deterministic to adaptive routing when the NoC is going to be congested.

The main purpose of this paper is to present a congestion control as the solution to a delay minimization problem for choosing the rate of BE sources. Our approach is different from the aforementioned works, e.g. [6][7], in which no delay consideration were taken into account. We present an algorithm as the solution to the optimization problem. To evaluate the performance of the proposed approach, we simulate the congestion control algorithm under a NoC-based scenario.

This paper is organized as follows. In Section 3 we present the system model and formulate the underlying optimization problem for BE congestion control. In Section 4 we solve the optimization problem using an iterative algorithm and propose the solution as a centralized congestion control algorithm to be implemented as a controller. In Section 5 we analyze the convergence behavior of the proposed algorithm and prove the underlying theorem of its convergence. In Section 6 we present the simulation results. Finally, the section 7 concludes the paper.

3. System Model

We consider a NoC with two dimensional mesh topology and wormhole routing. In wormhole networks, each packet is divided into a sequence of *flit*s which are transmitted over physical links one by one in a pipeline fashion. The NoC architecture is assumed to be lossless, and packets traverse the network on a shortest path using a deadlock free XY routing. We model the congestion control problem in NoC as the solution to an optimization problem. For more convenience, we turn the aforementioned NoC architecture into a mathematical model as in [9]. In this respect, we consider NoC as a network with a set of bidirectional links L and a set of sources S. A source consists of Processing Elements (PEs), routers and Input/Output ports. Each link $l \in L$ is a set of wires, busses and channels that are responsible for connecting different parts of the NoC and has a fixed capacity of c_l packets/sec. We denote the set of sources that share link l by $S(l)$. Similarly, the set of links that source s passes through, is denoted by $L(s)$. By definition, $l \in S(l)$ if and only if $s \in L(s)$.

As previously stated, there are two types of traffic in a NoC: GS and BE. For notational convenience, we divide S into two parts, each one representing sources with the same traffic. In this respect, we denote the set of sources with BE and GS traffic by S_{BE} and S_{GS}, respectively. Each link l is shared between the two aforementioned traffics. GS sources will obtain the required amount of the capacity of links and the remainder should be allocated to BE sources.

3.1. Delay Model

In recent years, researchers have presented different delay models in NoC (e.g. [10] and references therein). Due to simplicity of the model introduced in [10], we adopt its model in our framework.

Interconnects and network routers are two fundamental parts of the NoC which are subject to power consumption and communication latency. In our model, the delay of link $l \in L$ is denoted by d_l which represents the delay incurred to the system by packet propagation over this link. More precisely, d_l is given by

$$d_l = d_w + d_r \tag{1}$$

where d_w and d_r are delay of unit flow on interconnects and routers, respectively. In this respect, when a flow of amount f_l passes through link l, the total latency is:

$$D_l = f_l d_l \tag{2}$$

Interconnect or wire delay, d_w, is closely related to the wire styles. We assume that four types of wire styles are available for interconnects, namely, RC wires with repeated buffers with wire pitch varying from 1×, 2×, and 4× minimum global wire pitch, and on-chip transmission line with wire pitch equal to 16 micron. For RC wires with repeated buffers, we assume d_w is proportional to wire length, as below:

$$d_w = \text{per grid length delay} \times \text{wire length}$$

On the other hand, for on-chip transmission line, relatively large setup cost should be added to d_w. We use transmission line model proposed by Chen et al. [11] to estimate transmission line delay. Table 1 lists delay per grid length (2mm) of these four types of wire styles in 0.18 micron design technology. Setup cost of 50ps is added to d_w for transmission line.

We use the router delay model proposed by Peh et al. [12] to estimate NoC router delay. Table 2 shows

Table1: Delay Model of Wires

Wire Type	RC-1x	RC-2x	RC-4x	T-line
d_w (ns)	0.127	0.112	0.100	0.020

latency of routers in 0.18 micron technology node. When router input/output ports increase, d_r increases almost linearly.

Table 2: Model of Routers

Ports	2	3	4	5	6	7	8
d_r (ns)	0.599	0.662	0.709	0.756	0.788	0.819	0.835

3.2. Flow Control Model

Our objective is to choose source rates (IP loads) of BE traffics so that to minimize the sum of delays of all BE traffics. Hence the minimization problem can be formulated as:

$$\min_{x_s} \sum_{l=1}^{L} D_l \quad (3)$$

subject to:

$$\sum_{s\in S_{BE}(l)} x_s + \sum_{s\in S_{GS}(l)} x_s \le c_l \qquad \forall l \in L \quad (4)$$

$$\sum_{s\in S_{BE}} x_s \ge f \quad (5)$$

$$x_s > 0 \qquad \forall s \in S_{BE}$$

where source rates, i.e. x_s, $s \in S$, are optimization variables.

Regarding (2), we rewrite (3) as below:

$$\min_{x_s} \sum_{l=1}^{L} \Big(\sum_{s\in S_{BE}(l)} x_s\Big) d_l \quad (6)$$

subject to:

$$\sum_{s\in S_{BE}(l)} x_s + \sum_{s\in S_{GS}(l)} x_s \le c_l \qquad \forall l \in L \quad (7)$$

$$\sum_{s\in S_{BE}} x_s \ge f \quad (8)$$

$$x_s > 0 \qquad \forall s \in S_{BE}$$

The constraint (7) says that the aggregate BE source rates passing thorough link l cannot exceed its free capacity, i.e. the portion of the link capacity which has not been allocated to GS sources. The constraint (8) says that the sum of BE source rates must be at least f. For notational convenience, we define:

$$\hat{c}_l = c_l - \sum_{s\in S_{GS}(l)} x_s \quad (9)$$

Hence, (7) can be rewritten as:

$$\sum_{s\in S_{BE}(l)} x_s \le \hat{c}_l \qquad \forall l \in L \quad (10)$$

Although problem (6) can be separated across sources, its constraints will remain coupled across the network. Due to coupled nature of such constrained problems, they have to be solved using centralized methods like interior point methods [13]. Such computations may pose great overheads on the system. Instead of such methods, we seek to obtain the solution with simpler operations. One way is to use the *Projected Gradient Method* for constrained optimization problems [13] which will be briefly reviewed in the next section.

For notational convenience in solving the problem, we use matrix notation. In this respect, we define Routing matrix, i.e. $R = [R_{ls}]_{L\times S}$, as following:

$$R_{ls} = \begin{cases} 1 & \text{if } s \in S_{BE}(l) \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

We also define the source rate vector (for BE traffic), link delay and link capacity vectors as $x = (x_s, s \in S_{BE})$, $d = (d_l, l \in L)$ and $\hat{c} = (\hat{c}_l, l \in L)$, respectively. Therefore problem (6) can be rewritten in the matrix form as follows:

$$\min_{x_s} \; d^T R x \quad (12)$$

subject to:

$$Rx \le \hat{c} \quad (13)$$

$$\mathbf{1}^T x \ge f \quad (14)$$

$$x_s > 0 \qquad \forall s \in S_{BE}$$

where $\mathbf{1}$ is a vector with all one.

4. Congestion Control Algorithm

In this section, we will solve the optimization problem using Projected Gradient Method for constrained problems [13][14] and present a congestion control scheme for BE traffic in NoC systems to overcome the congestion.

The Projected Gradient Method for constrained minimization problems is very similar to the original one which only applies to unconstrained ones [13]. We briefly review this method in the following.

Consider the constrained minimization problem

$$\min_{x} f_o(x) \quad (15)$$

subject to:

$$f_i(x) \le 0, \qquad i = 1..m \quad (16)$$

in which $f_i : R^n \to R$ are convex functions.

In order to solve (15) iteratively, we define the following minimizing sequence

$$x^{(k+1)} = x^{(k)} - \gamma_k g^{(k)} \quad (17)$$

where

$$g^{(k)} \in \begin{cases} \nabla f_o(x) & f_j(x) \le 0, \quad i = 1,\dots,m \\ \nabla f_j(x(t)) & f_j(x) > 0 \end{cases} \tag{18}$$

and γ_k is the step size which satisfies:

$$\gamma_k > 0 \tag{19}$$

$$\gamma_k \to 0 \tag{20}$$

$$\sum_{k=1}^{\infty} \gamma_k = \infty \tag{21}$$

To quantify the performance of the method we define

$$f_{best}^k = \min\left\{ f_o(x^{(i)}) \middle| x^{(i)} \ feasible, \ i = 1,\dots,k \right\}$$

The following lemma, states the conditions on g under which the minimizing sequence (17) converges to the optimal point of (15), i.e. $f_{best}^k \to f^*$ and $x^k \to x^*$ as $k \to \infty$.

Lemma 1: *Consider the constrained minimization problem, as in (15). The minimizing sequence defined by (17) and (18) with the stepsize satisfying (19)-(21), converges to the optimal point of (15), i.e.* x^**, if the following conditions hold*

$$\left\| g^{(k)} \right\|_2 \le G \tag{22}$$

$$\left\| x^{(1)} - x^* \right\|_2 \le E \tag{23}$$

Proof: *See [14].*

In the sequel, we will solve the optimization problem (12) using Projected Gradient Method for constrained problems as stated in Lemma 1. Regarding (17), we have to calculate $g^{(k)}$. According to (18), if $x^{(k)}$ is feasible, i.e. $Rx \le \hat{c}$ and $\mathbf{1}^T x \ge f$, we have:

$$g = \nabla d^T R x = \nabla R^T d x = R^T d \tag{24}$$

otherwise at least one of the constraints must be violated. Assume link capacity constraint is violated for link l', i.e. $\sum_{s \in S_{BE}(l')} x_s > \hat{c}_{l'}$. Rewriting this in matrix form, yields:

$$\mathbf{e}_{l'}^T (Rx - \hat{c}) > \mathbf{0} \tag{25}$$

where $\mathbf{e}_{l'}$ is the l'th unit vector of R^L space which is zero in all entries except the l'th at which it is 1. Therefore, g is given by:

$$g = \nabla \mathbf{e}_{l'}^T (Rx - \hat{c}) = R^T \mathbf{e}_{l'} \tag{26}$$

Assuming that link capacity constraints are being satisfied, the sum-rate constraint is violated, i.e. $\mathbf{1}^T x < f$, or equivalently in the standard form as in (16), $f - \mathbf{1}^T x > 0$. Therefore g is given by:

$$g = -\nabla \mathbf{1}^T x = -\nabla x^T \mathbf{1} = -\mathbf{1} \tag{27}$$

Using (24), (26) and (27), the update equation to solve problem (12) is given by:

$$x_s^{(k+1)} = \left[x_s^{(k)} - \gamma_k g^{(k)} \right]^+ \tag{28}$$

where $[z]^+ = \max\{z, 0\}$ to satisfy non-negativity of source rate and $g^{(k)}$ is given by:

$$g^{(k)} = \begin{cases} R^T d & \sum_{s \in S_{BE}(l)} x_s(t) \le \hat{c}_l, \forall l \quad \text{and} \quad \mathbf{1}^T x \ge f \\ R^T e_{l'} & \sum_{s \in S_{BE}(l')} x_s(t) > \hat{c}_{l'}, \quad \exists l' \\ -\mathbf{1} & \mathbf{1}^T x \le f \end{cases} \tag{29}$$

(28) and (29) together propose an iterative algorithm as the solution to problem (12). In this respect, optimal source rates for BE sources can be found while satisfying capacity constraints and preserving GS traffic requirements. Thus, the aforementioned algorithm can be employed to control the congestion of the BE traffic in the NoC. The iterative algorithm is decentralized in the nature and can be addressed in distributed scenarios. However, due to well-formed structure of the NoC, we focus on a centralized scheme; we consider a controller to be mounted in the NoC to implement the proposed algorithm. The necessary requirement of such a controller is the ability to accommodate simple mathematical operations as in (28) and (29) and the allocation of few wires to communicate congestion control information to nodes with a light GS load. Algorithmic realization of proposed Congestion-Controller for BE traffic is listed below as Algorithm 1.

5. Convergence Analysis

In this section, we investigate the convergence analysis of the proposed algorithm using a time-varying stepsize in (28). As stated in the previous section, in this paper the stepsize is selected as (19)-(21).

Theorem 1: *The iterative congestion control scheme proposed by (28) and (29) with a time-varying stepsize which satisfies (19)-(21), will converge to the optimal point of problem (6).*

Proof: By lemma 1, it is clear that if its assumptions hold, the proof of Theorem is done. In this respect, $g^{(k)}$ should admit an upper bound in l_2-norm. In doing so, it suffices to show that its gradient is upper bounded in l_2-norm. Considering (29), we have

$$\left\|g^{(k)}\right\|_2 \le \max\{\|-\mathbf{1}\|_2, \left\|R^T d\right\|_2, \left\|R^T \mathbf{e}_{l'}\right\|_2\} \\ = S \tag{30}$$

Hence g in l_2-norm is bounded at least with S.

In the next step, we show that the Euclidian distance of the initial point to the optimal point is bounded at least with D, i.e.

$$\exists D > 0 \qquad \text{s.t.} \qquad \left\|x(1) - x^*\right\|_2 \le D \tag{31}$$

We have $x_s > 0, \ \forall s \in S_{BE}$. On the other hand, optimal source rates are bounded at most with maximum value of link capacities, i.e.

$$\max_s x_s^* \le \max_l \hat{c}_l \le \max_l c_l \tag{32}$$

Therefore,

$$\begin{aligned} \left\|x(1) - x^*\right\|_2 &\le \left\|\max x^* - \min x(1)\right\|_2 \\ &= \left\|\max_l c_l - 0\right\|_2 \\ &= Lc_{l\max} \end{aligned} \tag{33}$$

Hence the initial Euclidian distance is bounded with at least $Lc_{l\max}$. (30) and (33) complete the proof.

6. Simulation Results

In this section we examine the proposed congestion control algorithm, listed above as Algorithm 1, for a typical NoC architecture. We have simulated a NoC with 4×4 Mesh topology which consists of 16 nodes communicating using 24 shared bidirectional links; each one has a fixed capacity of 1 Gbps. In our scenario, packets traverse the network on a shortest path using a deadlock free XY routing. We also assume that each packet consists of 500 flits and each flit is 16 bit long.

One of the most significant issues of our interest, is the convergence behavior of the source rates. The step size is chosen to be $\gamma_k = 3/(1+k)$ which apparently satisfies (19)-(21). Variation of source rates for some nodes using above parameters are shown in Fig. 1.

Regarding Fig. 1, it's apparent that after about 270 iteration steps, all source rates will be in the vicinity of the steady state point of the algorithm; however, for the second case, at least 600 iteration steps is needed that

Algorithm 1: Congestion Control for BE Traffics in NoC

Initialization:

1. **Initialize** $\hat{c}_l$ **of all links.**
2. **Set source rate vector to zero.**

Loop:

Do until $(\max\left|x_s^{(k+1)} - x_s^{(k)}\right| < Error)$

1. $\forall s \in S$ **: Compute new source rate:**

$$x_s^{(k+1)} = \left[x_s^{(k)} - \gamma_k g^{(k)}\right]^+$$

where γ_k **can be selected as** $\gamma_k = a/(b+t)$ and

$$g^{(k)} = \begin{cases} R^T d & \sum_{s \in S_{BE}(l)} x_s(t) \le \hat{c}_l, \forall l \quad and \quad 1^T x \ge f \\ R^T e_{l'} & \sum_{s \in S_{BE}(l')} x_s(t) > \hat{c}_{l'}, \exists l' \\ -1 & 1^T x \le f \end{cases}$$

Output:

Communicate BE source rates to the corresponding nodes.

after which the source rates to be in the vicinity of the steady state point.

In order to have a better insight about the algorithm behavior, the relative error with respect to optimal rates which averaged over all sources, is also shown in Fig. 2. Optimal source rates are obtained using CVX [15] which is MATLAB-based software for solving disciplined convex optimization problems. As shown in Fig. 2, it is clear that after about 380 steps, the average of relative error of all sources falls below 20%, which is acceptable in practice. Thus, the proposed congestion control algorithm is computationally tractable.

Our final result is devoted to investigate the performance of algorithm 1 in terms of sum of delay in the network. In this respect, we have calculated sum of the delay for two cases; using Algorithm 1 and using uniform rate allocation. The result is depicted in Fig. 3. As a comparison, we conclude that the delay-sum is reduced at least by a factor of two which verifies the aim of the underlying optimization problem in source assignment in terms of delay-sum reduction.

7. Conclusion and Future Works

In this paper we addressed the problem of congestion control for BE traffic in NoC systems. Congestion control was modeled as the solution to the delay-sum minimization problem which was solved using gradient projection method for constrained

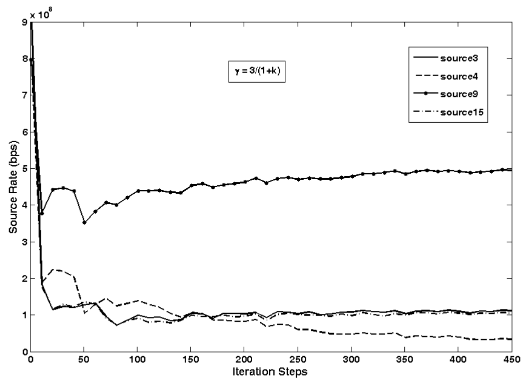

Fig. 1. Source rates for $\gamma_k = 3/(1+k)$

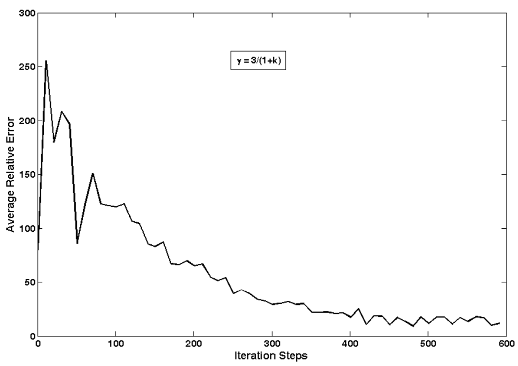

Fig. 2. Average of Error with respect to optimal solution for $\gamma_k = 3/(1+k)$

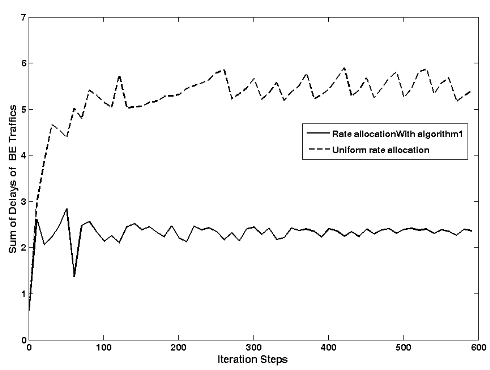

Fig. 3. Delay-Sum Comparison between proposed rate allocation and uniform rate allocation

optimization problems. This was led to an iterative algorithm which determine optimal BE source rates. We have also studied the realization of the algorithm as a centralized congestion controller. Simulation results confirm that the proposed algorithm converges and the computational overhead of the congestion control algorithm is small.

8. References

[1] L. Benini and G. DeMicheli, "Networks on Chips: A New SoC Paradigm." *Computer*, vol. 35, no. 1, pp. 70-78, 2002.

[2] F. P. Kelly, A. Maulloo and D. Tan, "Rate control for communication networks: Shadow prices, proportional fairness, and stability." *Operational Research Society*, vol. 49, no. 3, pp. 237–252, 1998.

[3] S. Mascolo. "Classical control theory for congestion avoidance in high-speed internet." *Decision and Control IEEE Conference*, vol. 3, pp. 2709-2714, 1999.

[4] Y. Gu, H. O. Wang and L.G. Yiguang Hong Bushnell, "A predictive congestion control algorithm for high speed communication networks." *American Control Conference*, vol. 5, pp. 3779-3780, 2001.

[5] C. Yang and A. V. S. Reddy, "A taxonomy for congestion control algorithms in packet switching networks." *IEEE Network*, vol. 9, no. 4, pp. 34-45, 1995.

[6] U. Y. Ogras and R. Marculescu, "Prediction-based flow control for network-on-chip traffic." *In Proceedings of the Design Automation Conference,* 2006.

[7] J. W. van den Brand, C. Ciordas, K. Goossensl and T. Basten, "Congestion-Controlled Best-Effort Communication for Networks-on-Chip." *Design, Automation and Test in Europe Conference*, pp. 948-953, 2007.

[8] J. Hu and R. Marculescu, "DyAD - smart routing for networks-on-chip." *Design Automation Conference*, pp. 260- 263, 2004.

[9] S. H. Low, D. E. Lapsley, "Optimization Flow Control, I: Basic Algorithm and Convergence", *IEEE/ACM Transactions on Networking*, vol. 7, no. 6, pp. 861-874, 1999.

[10] Y. Hu, Y. Zhu, H. Chen, R. Graham and C. K. Cheng, "Communication latency aware low power NoC synthesis." *Annual ACM IEEE Design Automation Conference*, pp. 574 – 579, 2006.

[11] H. Chen, R. Shi, C.K. Cheng, and D. Harris, "Suriner: A Distortionless Electrical Signaling Scheme for Speed of Light On-Chip Communications." *IEEE Intl. Conf. on Computer Design*, pp.497-502, 2005.

[12] L.S. Peh, "Flow Control and Micro-Architectural Mechanisms for Extending the Performance of Interconnection Networks." *Ph.D. Thesis, Stanford University*, 2001.

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

[14] S. Boyd, Convex Optimization II Lecture Notes, Stanford University, 2006.

[15] M. Grant, S. Boyd and Y. Ye, CVX (Ver. 1.0RC3): Matlab Software for Disciplined Convex Programming. Download available at: http://www.stanford.edu/~boyd/cvx.