An LP for Stabilization over Networks with Rate Constraints

Michael Rotkowitz[1,3]   Girish Nair[2,3]

Abstract

We consider the problem of stabilizing a network consisting of linear time-invariant plants, sensors, controllers, and relays, where the links can be rate-limited. A recent result shows how to characterize such networks for which stabilizing controllers exist, and then shows how to synthesize the coding and control laws to stabilize the network. This paper shows how that characterization can be expressed as an LP, and how that LP can then be extended to find coding/control laws which are optimal in a sense. It is further shown how to find such laws using a sparse portion of the network when possible.

1 Introduction

We consider a network of linear time-invariant plants, sensors, controllers, and relays. Each sensor gets measurements from a particular plant, and can then pass that on either to a relay, or directly to a controller, which would then give an input for a particular plant. Any of these links may be rate-limited. The framework for this paper mostly follows the development in [4], and that paper should be consulted for more detail on any of the setup, definitions, or assumptions which may be described in less detail here.

In that work, conditions were developed which characterized when such a network can be stabilized in terms of the available data rates, the eigenvalues of the plant modes, and the network topology. This was achieved using pseudorates, which corresponded to the amount of each channel being used to stabilize a particular mode, and then using these pseudorates to synthesize coding and control laws which stabilize the network. The pseudorates associated with every possible nontrivial irreducible cycle emanating from an unstable plant mode needed to satisfy certain inequalities.

In this paper, we show how, by generalizing some definitions, the conditions for stabilizability can be written as a Linear Program (LP), without any need to work out the nontrivial irreducible cycles connected to each mode. This is important because an LP can be solved with standard software, and can either find the global optimum or determine the problem to be infeasible efficiently in polynomial time.

Given this LP for stabilizability, we then consider how to choose from among the feasible psuedorates. We introduce two ways to optimize or enforce margins of error while maintaining an LP. We further show how to systematically choose sparse solutions, corresponding to using as few links as possible, from the remaining feasible solutions, again maintaining an LP.

2 Preliminaries

We consider the problem of stabilizing discrete-time linear time-invariant modes over a network of directed, point-to-point, errorless, finite bit-rate digital channels. The setup and notation closely follows that in [4]. A few aspects have been slightly generalized to lend itself to a more seamless transition to an LP in subsequent sections.

We now consider a network of N nodes. Each node can correspond to a plant mode, a controller, a sensor, or a relay. As in [4], we assume that each plant has real, distinct eigenvalues (though this can be relaxed), and thus that each plant node h can represent a 1-dimensional plant with dynamics matrix $\eta_h \in \mathbb{R}$.

With each node; that is, $\forall\ h \in 1, \ldots, N$, we associate a value H_h which can be interpreted as the entropy generated at that node.

$$H_h = \begin{cases} \max\{\log_2 |\eta_h|, 0\}, & \text{if node } i \text{ is a plant mode} \\ 0, & \text{otherwise.} \end{cases}$$

In other words, for an unstable plant mode this represents the log of the eigenvalue, or magnitude, and it is zero otherwise. If we were to generalize to multidimensional plants, this would become the sum of the logs of the unstable eigenvalues.

With each pair of nodes; that is, $\forall\ q, r \in 1, \ldots, N$, we associate a data rate $R_{q,r}$ as

$$R_{q,r} = \begin{cases} \text{average channel data rate,} & \\ & \text{if } q \text{ is a relay communicating to } r \\ & \text{if } q \text{ is a sensor communicating to } r \\ \infty, & \text{if } q \text{ is a controller affecting } r \\ & \text{if } r \text{ is sensing from } q \\ 0, & \text{otherwise.} \end{cases}$$

In other words, a generalized average data rate between nodes.

[1]Email: mcrotk@unimelb.edu.au
[2]Email: gnair@unimelb.edu.au
[3]Department of Electrical and Electronic Engineering, The University of Melbourne, Parkville VIC 3010 Australia

3 Stabilizability

Let $\mathcal{C}_h$ denote the set of all irreducible cycles which include a (plant) node h. For a given node h and cycle $c \in \mathcal{C}_h$, introduce a pseudorate $\tilde{\rho}_{h,c} \geq 0$.

Consider the following inequalities from [4]:

$$R_{q,r} \;\geq\; \sum_{c\in\mathcal{C}_h} \tilde{\rho}_{h,c} \quad \forall\, q, r \tag{1}$$

$$\sum_{c\in\mathcal{C}_h} \tilde{\rho}_{h,c} \;\geq\; \log_2|\eta_h| \tag{2}$$

The main result of [4] can be restated as such: feasibility of (1), (2) is necessary for the existence of controllers which uniformly stabilize the network, and strict feasibility is sufficient for the existence of such controllers.

This is sometimes paraphrased by saying that the inequalities are necessary and almost sufficient for stabilizability.

Let us now consider psuedorates $\rho_{h,i,j} \in \mathbb{R}$ associated with any triple of nodes $h, i, j \in 1, \ldots, N$. Feasibility of the above condition is then equivalent to feasibility of the following linear inequalities

$$R_{q,r} \;\geq\; \sum_{h} \rho_{h,q,r} \quad \forall\, q, r \tag{3}$$

$$\sum_{q} \rho_{h,q,r} \;\geq\; \sum_{s} \rho_{h,r,s} \quad \forall\, h, r \tag{4}$$

$$\sum_{q} \rho_{h,q,h} \;\geq\; H_h \quad \forall\, h \tag{5}$$

$$\rho_{h,i,j} \;\geq\; 0 \quad \forall\, h, i, j \tag{6}$$

The interpretation is that $\rho_{h,q,r}$ represents the amount of the data rate along the link $q \to r$ being used to stabilize mode h, but with our generalized definitions of H_h and $R_{q,r}$, we may consider them for any triple. The requirement in [4] that only nontrivial irreducible cycles emanating from unstable modes be considered is now automatically enforced.

4 Other Objectives

We now consider other qualities we may wish our pseudorates and corresponding stabilizing controller to have, and how to enforce them while retaining an LP.

Data rate buffer. We may wish to enforce a buffer of extra data rate along each link, in case there is some uncertainty regarding the available data rates, or to maximize such a buffer. This could be enforced by replacing (3) with the constraint

$$R_{q,r} - \sum_{h} \rho_{h,q,r} \;\geq\; \gamma^R \; R_{q,r} \quad \forall\, q, r \tag{7}$$

for some $\gamma^R \in (0, 1)$. Making the right-hand side multiplicative in $R_{q,r}$, rather than a constant, allows this to be applied seamlessly over links with a generalized data rate of zero or infinity. Using $\gamma^R \; \sum_h \rho_{h,q,r}$ for some $\gamma^R \in \mathbb{R}_+$ would work as well. Of course, γ^R could be made to vary from link to link as $\gamma^R_{q,r}$ if there are varying degrees of certainty regarding the available data rates.

Eigenvalue buffer. We may also or alternatively wish to enforce a buffer corresponding to the data rates available to stabilize each unstable mode, or to maximize such a buffer. This could be enforced by replacing (5) with

$$\sum_{q} \rho_{h,q,h} - H_h \;\geq\; \gamma^H \; H_h \quad \forall\, h \tag{8}$$

for some $\gamma^H \in \mathbb{R}^+$. Similar to above, making the right-hand side multiplicative in H_h, rather than a constant, allows this to be applied seamlessly over nodes with a generalized entropy of zero; that is, nodes which do not correspond to an unstable plant mode. Also similar to above, using $\gamma^H \; \sum_q \rho_{h,q,h}$ for some $\gamma^H \in (0, 1)$ would work as well, and γ^H could be made to vary from link to link as γ^H_h if there are varying degrees of certainty regarding the eigenvalues or varying requirements for disturbance attenuation. Recent results [3] show that, at least for simplified versions of this problem, the possible disturbance attenuation associated with this buffer can be an interpretable quantity related to a Bode integral.

Sparsity. Lastly, all else equal, we would typically like the resulting graph of (nonzero) pseudorates to be as sparse as possible; that is, for as many of the pseudorates to be zero as possible while stabilizing the network and satisfying the buffers. There are often inexplicit costs associated with using a link, such that, if a link is playing a very small role in stabilizing a node, and that node can be stabilized in another manner, we would typically prefer to not have to use that link at all.

Of course, directly minimizing the number of nonzero pseudorates would ruin our LP and lead to an intractable problem for all but the smallest networks. However, minimizing the 1-norm of a vector often yields sparse solutions and gives very good approximations for minimizing the cardinality. The reasons for this have become much better understood recently [2, 1].

We thus introduce the following objective for our optimization problem

$$\min \quad \|\mathrm{vec}(\rho)\|_1 \tag{9}$$

where $\mathrm{vec}(\rho)$ gives a vector in $\mathbb{R}^{N^3\times 1}$ containing all of the entries of $\rho_{h,q,r}$ for all $h, q, r \in 1, \ldots, N$.

5 Final LP

Choose $\alpha^S, \alpha^R, \alpha^H \in \mathbb{R}_+$ to represent the relative importance on sparsity, the rate buffer, and the eigenvalue buffer, respectively. They can be chosen w.l.o.g.

to sum to 1. We can now express our optimization problem as follows

$$\begin{aligned} \min \quad & \alpha^S \|\text{vec}(\rho)\|_1 - \alpha^R \gamma^R - \alpha^H \gamma^H \\ \text{subject to} \quad & R_{q,r} - \sum_h \rho_{h,q,r} \ \geq \ \gamma^R \ R_{q,r} \quad \forall\ q,r \\ & \sum_q \rho_{h,q,r} \ \geq \ \sum_s \rho_{h,r,s} \quad \forall\ h,r \\ & \sum_q \rho_{h,q,h} - H_h \ \geq \ \gamma^H \ H_h \quad \forall\ h \\ & \rho_{h,i,j} \ \geq \ 0 \quad \forall\ h,i,j \\ & \gamma^R \ \geq \ 0 \\ & \gamma^H \ \geq \ 0 \end{aligned}$$

Of course, if we instead wanted to impose a hard constraint for one of the buffers, rather than incorporating it into the objective function, we would just fix the corresponding γ and then remove the corresponding term from the objective. If we didn't need a particular buffer at all, we would just fix that γ to be 0.

We can now put this optimization problem in the standard LP form

$$\begin{aligned} \min \quad & c^T x \\ \text{subject to} \quad & Ax \ \leq \ b \end{aligned}$$

to be solved with, for example, the linprog() function in Matlab.

Let

$$\begin{aligned} x &= \begin{bmatrix} \text{vec}(\rho)^T & \gamma^R & \gamma^H \end{bmatrix}^T \in \mathbb{R}^{(N^3+2)\times 1} \\ c &= \begin{bmatrix} \alpha^S & \cdots & \alpha^S & -\alpha^R & -\alpha^H \end{bmatrix}^T \in \mathbb{R}^{(N^3+2)\times 1} \end{aligned}$$

Note that since the pseudorates are nonnegative, the usual trick for optimizing a 1-norm as an LP is not necessary. The form of A, B will depend on the exact choice of vec : $\mathbb{R}^{N\times N\times N} \to \mathbb{R}^{N^3\times 1}$ and is left to the reader. It then remains to replace the values of ∞ with an appropriately large number such that the result will be unaffected; anything larger than $(1+\gamma^H)(\sum_h H_h)/(1-\gamma^R)$, for the largest values of γ^H and γ^R likely to be considered, will suffice. After solving the standard LP, the optimal pseudorates can then be recovered as $\rho^* = \text{vec}^{-1}(\begin{bmatrix} I_{N^3} & 0 \end{bmatrix} x^*)$, which can then be used as described in [4] to synthesize the (optimal) stabilizing coding and control laws.

Acknowledgments

The first author would like to thank (Isaac) Chung-Yao Kao and Michael Cantoni for hosting him on his first visit to The University of Melbourne, during which this collaboration began.

6 Conclusions

Given a network of linear time-invariant plants, sensors, controllers, and relays, we have leveraged recent results to develop an LP to determine stabilizability of the network. We modified this LP to enforce or maximize margins of error, and further modified it to stabilize the network using a sparse potion of the available links when possible.

Interesting future work could include developing an understanding of the relationship between the buffers γ^R, γ^H and more traditional notions of stability margin.

References

[1] E.J. Candes and T. Tao. The Dantzig selector: statistical estimation when p is much larger than n. *Annals of Statistics*, to appear.

[2] D. Donoho. For most large underdetermined systems of linear equations the minimal ℓ^1-norm solution is also the sparsest solution. Technical Report 2004-9, Department of Statistics, Stanford University, July 2004.

[3] N.C. Martins and M.A. Dahleh. Feedback control in the presence of noisy channels: "Bode-like" fundamental limitations of performance. *IEEE Transactions on Automatic Control*, to appear, 53(5), May 2008.

[4] G.N. Nair and R.J. Evans. Cooperative networked stabilisability of linear systems with measurement noise. In *Proc. 15th IEEE Mediterranean Conf. Control and Automation*, 2007.