Deployment Versus Data Retrieval Costs for Caches in the Plane

Mihaela Mitici, Jasper Goseling, Maurits de Graaf, and Richard J. Boucherie

***Abstract*—We consider the problem of finding the Pareto front of the expected deployment cost of wireless caches in the plane and the expected retrieval cost of a client requesting data from the caches. The data is stored at the caches according to partitioning and coding strategies. We show that under coding, it is optimal to deploy many caches with low storage capacity. For partitioning, we derive a simple relation between the cost of the cache deployment and the cost of retrieving the data from the caches. We quantify the improvements offered by optimal coding in comparison to partitioning, i.e., we derive a relation for the difference in deployment cost required to achieve a given retrieval cost. Finally, we show that even non-optimal coding is better than partitioning in the sense that no coded deployment can be dominated by a partitioning strategy.**

***Index Terms*—Content distribution networks, network coding, pareto optimization.**

I. Introduction

WE CONSIDER wireless caches placed in the plane according to a homogeneous Poisson process. A client arriving at a random location in the plane is interested in retrieving a large data file that is stored at the caches. Since the storage capacity of the caches is limited, the file needs to be stored in a distributed fashion. Thus, the client needs to retrieve data fragments from several caches to recover the complete file.

Data fragments can be stored at the caches according to various strategies. We study two storage strategies: partitioning and coding [1]. Partitioning is a storage strategy according to which the data is divided into equally-sized fragments. Replicas of the data fragments are stored at the caches. In the coding strategy, each cache stores a random linear combination of the fragments.

We focus on two cost measures and their Pareto front. The first cost measure is the deployment cost of the caches in the plane, defined to be proportional to the storage capacity of the caches and the density of the Poisson process according to which the caches are deployed. The second cost measure is the cost for a client to retrieve the file from the caches. This is the cumulative cost of obtaining individual data fragments from specific caches. The retrieval cost is increasing in the distance between the client and the contacted caches. Fig. 1 illustrates how the retrieval cost is affected by the density of the caches. One could reduce the retrieval cost by increasing the density of the caches, i.e., the average number of caches per unit area. In this case, however, the deployment cost would increase. A trade-off arises between the deployment of the caches in the plane and the cost of retrieving the data. In the current work, we analyze the Pareto front of deployment and data retrieval costs under partitioning and coding. The parameters over which we optimize are: i) the density of the Poisson process according to which caches are deployed and ii) the storage capacity of an individual cache.

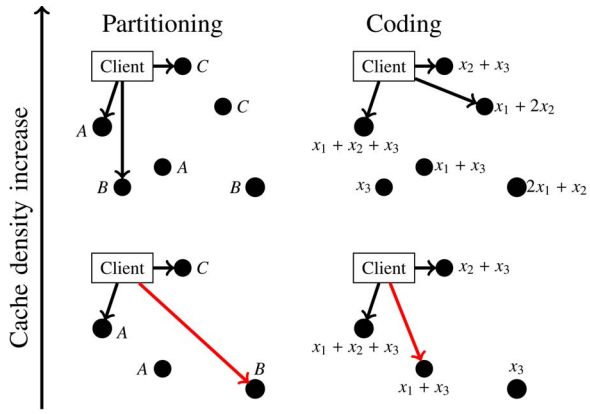

Fig. 1. Influence of data storage strategy and cache density on retrieval and deployment costs. Partitioning: the file consists of fragments A, B, C. Coding: fragments are independent linear combinations of x_1, x_2, x_3. High cache density enables shorter client-cache distances. Thus, the retrieval cost decreases at the expense of higher deployment costs.

Data replication and coding for caches have been studied in [2]–[10]. In [2], the authors consider the optimal number of replicas of data such that the distance between a node requesting data and the nearest replica is minimized. In [3], [4], the problem of which files to store at which caches based on the files' popularity is analyzed. A client requests content from a neighborhood of caches. The aim is to maximize the probability that a client's request is satisfied given the available files in its neighborhood of caches. Data sharing among multiple caches such that the bandwidth consumption and the data retrieval delay are minimal, is considered in [5]. In [6], the optimal collaboration distance between caches is analyzed. In [7] coded data allocation at the caches is investigated such that any sufficiently large subset of caches can provide the complete data. In [8] coding for networks of caches are presented, where each user has access to a single cache and a direct link to the

Manuscript received February 6, 2014; revised April 4, 2014; accepted April 15, 2014. Date of publication April 25, 2014; date of current version August 20, 2014. This work was performed within the project Realisation of Reliable and Secure Residential Sensor Platforms (RRR) of the Dutch program IOP Generieke Communicatie, IGC1020, supported by the Subsidieregeling Sterktes in Innovatie, and is supported in part by the Netherlands Organisation for Scientific Research (NWO) Grant 612.001.107. The associate editor coordinating the review of this paper and approving it for publication was W. Chen.

M. Mitici and R. J. Boucherie are with the Department of Applied Mathematics, University of Twente, 7522 NB Enschede, The Netherlands (e-mail: m.a.mitici@utwente.nl; r.j.boucherie@utwente.nl).

J. Goseling is with the Department of Applied Mathematics, University of Twente, 7522 NB Enschede, The Netherlands, and also with Faculty of Electrical Engineering, Mathematics and Computer Science (EEMCS), Delft University of Technology, 2628 CD Delft, The Netherlands (e-mail: j.goseling@utwente.nl).

M. de Graaf is with the Department of Applied Mathematics, University of Twente, 7522 NB Enschede, The Netherlands, and also with Thales B.V. Nederland, 7550 GD Hengelo, The Netherlands (e-mail: m.deGraaf@utwente.nl).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LWC.2014.2320512

source. It is shown how coding reduces the load on the link between the caches and the source. The impact of non-orthogonal transmissions for coding is considered in [9]. Scaling results on the best achievable transmission rates are derived. In this paper, we assume that data transmissions from the caches to the client are orthogonal, by separating them in time, frequency or coding space.

Cache deployment in the plane under the partitioning and coding storage strategies has been studied in [11]. It is shown that, for fixed cache density and cache storage capacity, coding dominates partitioning, i.e., the retrieval cost under coding is always smaller than under partitioning. Our interest is in optimizing the density and the capacity of the caches. This has motivated us to investigate in [12] the Pareto front of the cache deployment and data retrieval costs. We provided in [12] an approximate characterization of this Pareto front. In particular, we restricted the density to a fixed interval. Also, we relaxed the integrality condition on the cache storage capacity, i.e., we did not impose the constraint that the capacity is a integer multiple of the symbol size.

In *this paper* we provide an exact characterization of the Pareto fronts under the coding and partitioning strategies. The integrality constraint on the storage capacity is taken into account and we do not enforce a bound on the density of the caches. As a consequence, we obtain the complete Pareto fronts, whereas only part of these fronts were given in [12]. In addition, it is rigorously demonstrated that all points on the fronts can be achieved with integral storage capacity.

We show that for any Pareto optimal partitioning strategy there exists an optimal coding strategy that dominates it. We also show that under the partitioning strategy, the Pareto front depends only on the ratio of the deployment density and the storage capacity of the caches. Thus, when deploying caches, one has some flexibility in either increasing the density or the capacity of the caches. For the coding strategy, however, we show that it is optimal to deploy many caches with low storage capacities. Thus, even though the optimal coded strategy leads to lower costs, it provides less flexibility in deployment. We quantify the improvements offered by the optimal coding in comparison to the partitioning strategy, i.e., we derive a relation for the difference in deployment cost required to achieve a given retrieval cost. Finally, we show that even non-optimal coding is better than the partitioning strategy in the sense that no coded deployment can be dominated by a partitioning strategy.

II. Problem Statement

We consider a data file of n symbols, with $n > 1$ fixed, which is stored at the caches. The symbols are elements of a finite field $\mathbb{F}_q$.

Caches are placed in the plane according to a homogeneous Poisson process with density λ, where $\lambda > 0$ is a parameter over which we optimize. The caches have limited storage capacity which we express in terms of an integer k, the second optimization parameter. A cache stores n/k symbols. To ensure integral cache capacity, we impose that k divides n, denoted by $k|n$. Thus, $1 \leq k \leq n$ and n/k is an integer.

A client arriving at a random location in the plane is interested in retrieving the file from the caches. We assume that the client has complete knowledge about the content and the location of the caches. The client requests data from a set of k caches that ensure the recovery of the file.

Under the partitioning strategy (P), the file is divided into k different fragments, each of n/k symbols. Each cache selects uniformly at random a fragment to store. A client requests fragments from k closest caches such that all k fragments are distinct.

Under coding (C), each cache stores a random linear combination of the k fragments. The closest set of k caches is chosen to decode the file. There is a positive probability that k random linear combinations are not linearly independent and, therefore, do not provide the entire data. In this case, the client needs to request data from caches that are located further away. It was demonstrated in [11] that this has a negligible impact on the retrieval cost. Therefore, we restrict our attention to the cost of retrieving the data from the k nearest caches.

The cost measures, which are a function of the model parameters k and λ, are defined as follows:

i) The expected data retrieval cost, denoted by C_r^A, with $A \in \{P, C\}$.

Let the cost of retrieving data from k caches located at distances $\delta_1, \ldots, \delta_k$ be

$$\mathbf{C}_r^A(\delta_1, \ldots, \delta_k) = \frac{1}{n} \sum_{i=1}^{k} \frac{n}{k} \delta_i^{2\alpha} \tag{1}$$

where $\alpha \geq 1/2$ is an arbitrary, but fixed, parameter. This cost function can denote, for example, the power needed for a cache to transmit data to a client. Indeed, from the capacity of a AWGN channel $1/2 \log(1 + P\delta^{-2\alpha})$, where 2α is the path loss exponent in the wireless medium, the power to transmit at a guaranteed minimum rate R is $P = (e^{2R} - 1)\delta^{2\alpha}$.

In (1), $(n/k)\delta_i^{2a}$ is the cost of retrieving n/k symbols from a cache at distance δ_i away from the client. We normalize the retrieval cost by n. We are interested in the expected cost $C_r^A(k, \lambda)$, where the expectation is over the randomness in the spatial Poisson process.

ii) The expected deployment cost of the caches in the plane per unit area, denoted by C_d.

The cost of deploying a single cache is proportional to the cost of storing n/k symbols and the deployment density $\lambda > 0$ of the caches in the plane. We normalize the cost by n. The expected deployment cost per symbol per unit area is defined

$$C_d(k, \lambda) = \frac{\lambda}{k} \tag{2}$$

with the expectation over the randomness in the spatial Poisson process and the randomness of the storage strategy.

We consider the multi-objective optimization problem which aims at minimizing the expected deployment cost $C_d(k, \lambda)$ and the expected retrieval cost $C_r^A(k, \lambda)$ under the storage strategy $A \in \{P, C\}$.

We will make use of the gamma function, which for $x > 0$ is represented as $\Gamma(x) = \int_0^\infty t^{x-1} e^{-t} dt$ and the digamma function, $\psi(x) = (d/dx) \ln \Gamma(x) = \int_0^\infty ((e^{-t}/t) - (e^{-xt}/(1 - e^{-t})))dt$ (see, for instance, [13]).

III. Analysis

In general, a single point simultaneously minimizing two conflicting objectives does not exist, in which case the multi-objective problem does not have a unique optimal solution. Thus, we characterize the Pareto front [14] of the expected cache deployment and data retrieval costs. More precisely, we analyze which sets of objective values are achievable and non-dominated. Multi-objective methods such as scalarization or goal programming are commonly used to compute the Pareto front. We refer to [15] for an extensive survey on methods to compute optimal Pareto solutions. In this paper, the specific structure of the objective functions allows us to directly determine their Pareto front.

A. Partitioning

Theorem 1 [11]: The expected cost of retrieving the data file under the partitioning (P) strategy is

$$C_r^P(k,\lambda) = \left(\frac{k}{\lambda\pi}\right)^{\alpha} \Gamma(\alpha+1). \tag{3}$$

Our first contribution is the following result:

Theorem 2: The Pareto front of the partitioning strategy is described by the following set of points:

$$\left\{(x,y)\,|\,x>0,\ y=\frac{\Gamma(\alpha+1)}{(\pi x)^{\alpha}}\right\}.$$

Proof: The proof follows from writing the expected retrieval cost (3) as a function of the expected deployment cost (2).

The above results demonstrate that the Pareto front of the costs $C_r^P(k,\lambda)$ and $C_d^P(k,\lambda)$ only depends on the ratio of the optimization parameters $x=\lambda/k$.

B. Coding

Theorem 3 [11]: The expected cost of retrieving the data file under the coding (C) strategy is

$$C_r^C(k,\lambda) = \frac{1}{k}\left(\frac{1}{\lambda\pi}\right)^{\alpha}\frac{\Gamma(\alpha+1+k)}{(\alpha+1)\Gamma(k)}. \tag{4}$$

We first state the following lemma, which appears in [12].

Lemma 1 [12]: $v\psi(v+s)-v\psi(v)-s<0$, for $1\le v\le n$ and $s>1$.

Theorem 4: The Pareto front of the coding strategy is described by the following set of points:

$$\left\{(x,y)\,|\,x>0,\ y=\frac{\Gamma(\alpha+1+n)}{(\alpha+1)\Gamma(n)(\pi x)^{\alpha}n^{\alpha+1}}\right\}.$$

Proof: Let $x=C_d(k,\lambda)$, with $x>0$, and $y=C_r^C(k,\lambda)$. Then, using Theorem 3, the expected retrieval cost as a function of the expected deployment cost is as follows:

$$y=\frac{\Gamma(\alpha+1+k)}{(\alpha+1)(\pi x)^{\alpha}\Gamma(k)k^{(1+\alpha)}}. \tag{5}$$

Let

$$g(k,\alpha)=\frac{\Gamma(\alpha+1+k)}{(\alpha+1)\pi^{\alpha}\Gamma(k)k^{(1+\alpha)}}.$$

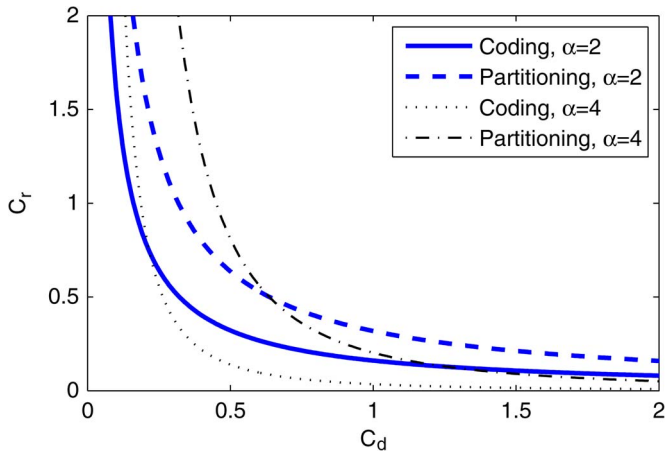

Fig. 2. The Pareto front for coding and partitioning, $\alpha=2$.

Then $y=g(k,\alpha)x^{-\alpha}$. We now show that $g(k,\alpha)$ is minimized when $k=n$. Computing the gradient of $g(k,\alpha)$ with respect to k gives

$$\frac{\Gamma(\alpha+1+k)}{(\alpha+1)\pi^{\alpha}\Gamma(k)k^{(2+\alpha)}}\left[k\psi(k+1+\alpha)-k\psi(k)-1-\alpha\right]$$

which by Lemma 1 is negative. It follows that $g(k,\alpha)$ is decreasing in k. Therefore, y is minimized for $k=n$.

Taking $k=n$ in (5) gives the desired result.

The above results show that under coding, it is Pareto optimal to always take $k=n$, i.e., to fragment the data as much as possible. Note that this result satisfies the integrality constraint of the storage capacity of the caches.

Fig. 2 shows the Pareto front of the deployment and retrieval costs under partitioning and coding. The figure shows that optimizing for one of the objectives necessarily influences the other. For example, a low data retrieval cost for the client can be achieved at the expense of a more dense, yet more expensive, cache deployment.

C. Performance Comparison Partitioning-Coding

It was shown in [12] that any point on the partitioning Pareto front can be achieved by parameters $k=n$ and a corresponding cache density λ^*. Also, it was shown that for each such point on the Pareto front there exists a point on the optimal coding Pareto front achieved by parameters $k=n$ and λ^* which is dominating. Thus, any point on the partitioning Pareto front is dominated by a point on the optimal coding Pareto front.

In this subsection we quantify the performance gap between the optimal coding and the partitioning strategies. In particular, we consider for a given expected retrieval cost y the difference $\Delta_d(y)$ of the optimal expected deployment cost under optimal coding and the optimal expected deployment cost under partitioning. More precisely, let the pairs (k_P,λ_P) and (n,λ_C) be Pareto optimal under partitioning and coding, respectively. Moreover, let $C_r^P(k_P,\lambda_P)=C_r^C(n,\lambda_C)=y$. Then

$$\Delta_d(y)=C_d^P(k_P,\lambda_P)-C_d^C(n,\lambda_C). \tag{6}$$

Similarly, we introduce the corresponding performance gap for the deployment costs

$$\Delta_r(x)=C_r^P(k_P,\lambda_P)-C_r^C(n,\lambda_C) \tag{7}$$

for the case that (k_P, λ_P) and (n, λ_C) are optimal and satisfy $C_d^P(k_P, \lambda_P) = C_d^C(n, \lambda_C) = x$.

Theorem 5: The performance gap between optimal partitioning and coding is

$$\Delta_r(x) = b_r x^{-\alpha}, \quad \Delta_d(y) = b_d y^{-1/\alpha}$$

where

$$b_r = \pi^{-\alpha}\Gamma(\alpha+1)\left(1 - f(n,\alpha)\right) \tag{8}$$

$$b_d = \pi^{-1}\Gamma(\alpha+1)^{\frac{1}{\alpha}}\left(1 - f(n,\alpha)^{1/\alpha}\right) \tag{9}$$

$$f(n,\alpha) = \frac{\Gamma(\alpha+1+n)}{n^{\alpha}\Gamma(\alpha+2)\Gamma(n+1)}. \tag{10}$$

Moreover,

$$0 \le b_r \le \pi^{-\alpha}\Gamma(\alpha+1)\left(1 - \Gamma(\alpha+2)^{-1}\right), \tag{11}$$

$$0 \le b_d \le \pi^{-1}\Gamma(\alpha+1)^{1/\alpha}\left(1 - \Gamma(\alpha+2)^{-1/\alpha}\right). \tag{12}$$

Proof: The results follow from (3), (4), the fact that $f(n,\alpha)$ is decreasing in n, $\lim_{n\to\infty} f(n,\alpha) = \Gamma(\alpha+2)^{-1}$ and $\lim_{n\to 1} f(n,\alpha) = 1$.

For a large-path loss exponent, which is the case in, for instance, millimeter-wave based communications, the performance gap between the coding and the partitioning strategies with respect to the retrieval cost can be arbitrarily large since $\lim_{\alpha\to\infty}\Delta_r(y) = \lim_{\alpha\to\infty}\Gamma(\alpha+1)^{1/\alpha}y^{-1/\alpha}\pi^{-1}n^{-1} = \infty$. In the case of the deployment cost, the performance gap is:

$$\lim_{\alpha\to\infty}\Delta_d(y) = \lambda_C\left(1 - \frac{1}{n}\right), \text{ where } y = C_r^C(n,\lambda_C).$$

The results follow from (3), (4) and $\lim_{\alpha\to\infty} f(n,\alpha)^{1/\alpha} = 1/n$.

D. Non-Optimal Coding vs. Partitioning

We next show that there exists no domain of the optimization parameters where partitioning dominates coding in terms of retrieval and deployment costs, even with a non-optimal parameter choice. More precisely, consider the coding strategy and allow the deployment to be non-optimal, i.e., we consider deployment of caches with parameters λ_C and k_C, where we allow $k_C < n$. We show that there exists no deployment scenario, i.e., no values λ_P, k_P, under which partitioning dominates coding.

Theorem 6: There exists no values λ_P, λ_C and $1 < k_P$, $k_C \le n$ such that $C_d^P(k_P,\lambda_P) \le C_d^C(k_C,\lambda_C)$ and $C_r^P(k_P, \lambda_P) \le C_r^C(k_C,\lambda_C)$, with at least one of these inequalities holding strictly.

Proof: From $C_d^P(k_P,\lambda_P) \le C_d^C(k_C,\lambda_C)$ and $C_r^P(k_P, \lambda_P) \le C_r^C(k_C,\lambda_C)$ it follows by definition of the deployment cost that

$$1 \le \frac{k_P\lambda_C}{k_C\lambda_P} \tag{13}$$

and from (3) and (4) that

$$\frac{k_P\lambda_C}{k_C\lambda_P} \le \left(\frac{\Gamma(\alpha+1+k_C)}{\Gamma(k_C+1)k_C^{\alpha}\Gamma(\alpha+2)}\right)^{1/\alpha} \tag{14}$$

respectively. Strict inequality in (13) and (14) holds only if $C_d^P(k_P,\lambda_P) < C_d^C(k_C,\lambda_C)$ and $C_r^P(k_P,\lambda_P) < C_r^C(k_C, \lambda_C)$, respectively. The result follows from the observation that

$$\frac{\Gamma(\alpha+1+k)}{k^{\alpha}\Gamma(\alpha+2)\Gamma(k+1)} < 1, \quad \text{for any } k > 1.$$

IV. Conclusion

We determined the Pareto front of the expected deployment cost of the caches in the plane and the expected cost for a client to retrieve a large data file from the caches. The Pareto front shows to what extent one of the objectives can be improved at the expense of the other. For partitioning, we derived a simple relation for the Pareto points. For coding, we showed that it is Pareto optimal to maximize the data fragmentation. We showed that storing data according to the coding strategy results in a lower Pareto front than in the case of partitioning. We also quantified the additional cost incurred by partitioning in comparison to optimal coding when a specific retrieval or deployment cost is given. Lastly, we showed that even non-optimal coding is better than the partitioning strategy, i.e., no coded deployment can be dominated by a partitioning strategy.

References

[1] M. Médard and A. Sprintson, *Network Coding: Fundamentals and Applications*. New York, NY, USA: Academic, 2012.

[2] S. Jin and L. Wang, "Content and service replication strategies in multi-hop wireless mesh networks," in *Proc. Int. Symp. Model., Anal. Simul. Wireless Mobile Syst.*, 2005, pp. 79–86.

[3] N. Golrezaei, K. Shanmugam, A. Dimakis, A. Molisch, and G. Caire, "Femtocaching: Wireless video content delivery through distributed caching helpers," in *Proc. IEEE INFOCOM*, 2012, pp. 1107–1115.

[4] N. Golrezaei, A. Molisch, A. Dimakis, and G. Caire, "Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution," *IEEE Commun. Mag.*, vol. 51, no. 4, pp. 142–149, Apr. 2013.

[5] L. Yin and G. Cao, "Supporting cooperative caching in ad hoc networks," *IEEE Trans. Mobile Comput.*, vol. 5, no. 1, pp. 77–89, Jan. 2006.

[6] N. Golrezaei, A. Molisch, and A. Dimakis, "Base-station assisted device-to-device communications for high-throughput wireless video networks," in *Proc. IEEE Int. Conf. Commun.*, 2012, pp. 7077–7081.

[7] A. G. Dimakis, V. Prabhakaran, and K. Ramchandran, "Ubiquitous access to distributed data in large-scale sensor networks through decentralized erasure codes," in *Proc. Int. Symp. Inf. Process. Sensor Netw.*, 2005, p. 15.

[8] M. Maddah-Ali and U. Niesen, "Fundamental limits of caching," *IEEE Trans. Inf. Theory*, vol. 60, no. 5, pp. 2586–2867, 2014.

[9] U. Niesen, D. Shah, and G. Wornell, "Caching in wireless networks," *IEEE Trans. Inf. Theory*, vol. 58, no. 10, pp. 6524–6540, Oct. 2012.

[10] A. Dimakis, P. Godfrey, Y. Wu, M. Wainwright, and K. Ramchandran, "Network coding for distributed storage systems," *IEEE Trans. Inf. Theory*, vol. 56, no. 9, pp. 4539–4551, Sep. 2010.

[11] E. Altman, K. Avrachenkov, and J. Goseling, "Distributed storage in the plane," presented at the IFIP Networking Conference, Trondheim, Norway, Jun. 2014.

[12] M. Mitici, J. Goseling, M. de Graaf, and R. J. Boucherie, "Optimal deployment of caches in the plane," in *Proc. IEEE Global Signal Inf. Process.*, 2013, pp. 863–866.

[13] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions: With Formulas, Graphs, and Mathematical Tables*. New York, NY, USA: Dover, 1964.

[14] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[15] G. Liu, J. Whidborne, and J.-B. Yang, *Multiobjective Optimisation and Control*. Baldock, U.K.: Research Studies Press, 2003.