

# A Sequential Constraint Relaxation Algorithm for Rank-One Constrained Problems

Pan Cao\*, John Thompson\*, H. Vincent Poor†

\*Institute for Digital Communications, The University of Edinburgh, EH9 3JL, UK

†Department of Electrical Engineering, Princeton University, NJ 08544, USA

Email: \*{p.cao, john.thompson}@ed.ac.uk, †poor@princeton.edu

**Abstract**—Many optimization problems in communications and signal processing can be formulated as rank-one constrained optimization problems. This has motivated the development of methods to solve such problem in specific scenarios. However, due to the non-convex nature of the rank-one constraint, limited progress has been made in solving generic rank-one constrained optimization problems. In particular, the problem of efficiently finding a locally optimal solution to a generic rank-one constrained problem remains open. This paper focuses on solving general rank-one constrained problems via relaxation techniques. However, instead of dropping the rank-one constraint completely as is done in traditional rank-one relaxation methods, a novel algorithm that gradually relaxes the rank-one constraint, termed the sequential rank-one constraint relaxation (SROCR) algorithm, is proposed. Compared with previous algorithms, the SROCR algorithm can solve general rank-one constrained problems, and can find feasible solutions with favorable complexity.

## I. INTRODUCTION

Recently there has been a growing interest in optimization problems that involve rank-one constraints. This is because many optimization problems in communications and signal processing applications can be cast as such problems [1].

Consider the following rank-one constrained optimization problem:

$$\min_{\mathbf{X} \succeq \mathbf{0}} g_0(\mathbf{X}) \quad (1a)$$

$$\text{s.t. } g_k(\mathbf{X}) \triangleleft_k b_k, \quad k = 1, \dots, K \quad (1b)$$

$$\text{rank}(\mathbf{X}) = 1, \quad (1c)$$

where  $g_0, g_1, \dots, g_K : \mathbb{C}^{N \times N} \rightarrow \mathbb{R}$  are continuous and differentiable convex or affine functions of an  $N \times N$  complex-valued positive semidefinite matrix variable  $\mathbf{X} \succeq \mathbf{0}$ ,  $b_1, \dots, b_K \in \mathbb{R}$  are scalar constants, and " $\triangleleft_k$ " in (1b) could refer to " $<$ ", " $\leq$ " for the  $k$ -th constraint in (1b), or " $=$ " for linear  $g_k(\mathbf{X})$ . We assume that the constraints (1b) and  $\mathbf{X} \succeq \mathbf{0}$  form a bounded convex set to ensure that the objective function  $g_0(\mathbf{X})$  always achieves a finite value that is meaningful in engineering applications.

Without the rank-one constraint  $\text{rank}(\mathbf{X}) = 1$ , Problem (1)

becomes the following relaxed problem:

$$\min_{\mathbf{X} \succeq \mathbf{0}} g_0(\mathbf{X}) \quad (2a)$$

$$\text{s.t. } g_k(\mathbf{X}) \triangleleft_k b_k, \quad k = 1, \dots, K, \quad (2b)$$

which now is a convex optimization problem and thus can be efficiently solved by standard convex optimization methods, e.g., CVX. However, since the rank function is quasi-concave and subadditive [2], the non-convex constraint  $\text{rank}(\mathbf{X}) = 1$  causes the original problem (1) to become NP-hard in general. In some special cases, Problem (1) may be convex or otherwise have lower complexity [3], [4]. However, in this work, we will focus on the generic case when Problem (1) is NP-hard.

With this viewpoint, some heuristic approaches have been proposed to obtain sub-optimal or feasible rank-one solutions for Problem (1). These can be categorized into several approaches: 1) Rank-one constraint relaxation – first solve the relaxed problem without the rank-one constraint, and then construct a rank-one solution based on the solution of the relaxed problem if its solution is not rank-one. Otherwise, the solution of the relaxed optimization problem is exactly the solution to the original rank-one constrained problem [1], [5]–[8]. A drawback of this approach is that solving the relaxed problem usually yields a high rank solution, and in addition the constructed rank-one solution is usually sub-optimal or even infeasible for the rank-one constrained problem; 2) Rank minimization – first approximate the rank function by some tractable function, and then add the approximate rank function to the original objective function as a penalty term [2], [9], [10]. A difficulty with this approach is that its performance depends on both the quality of the rank-function approximation and the penalty parameter setting, as it is well known that different penalty parameters influence both the performance and complexity of penalty-based algorithms; 3) Quadratic optimization – first rewrite a rank-one positive semidefinite matrix variable as a quadratic term of a vector variable such that the rank-one constraint can be avoided, and then solve a quadratic optimization problem to obtain a vector variable [11], [12]. Unfortunately, there is no "free lunch", since in general it is still hard or even more difficult to solve a quadratic optimization problem. Thus, it is common to use semidefinite programming (SDP) to formulate a quadratic optimization problem as a rank-one constrained optimization problem.

These existing heuristic approaches typically work for some

The work of P. Cao and J. Thompson was supported by the UK EPSRC under grant EP/L026147/1. The work of H. Poor was supported in part by the U. S. National Science Foundation under Grants CCF-1420575 and ECCS-1343210.

specific cases but not generic problems. This motivates us to propose a novel rank-one constrained optimization technique – sequential rank-one constraint relaxation (SROCR). Instead of dropping the rank-one constraint completely, the basic idea of this algorithm is to relax the rank-one constraint gradually such that it is easier to find a feasible solution. In addition, the SROCR algorithm enables one to generate a locally optimal solution to a generic rank-one constrained optimization problem if the relaxed optimization problem without a rank-one constraint is a convex optimization problem. The proposed algorithm is evaluated via numerical results, which imply that the proposed algorithm can usually achieve a better objective function value with lower or comparable complexity compared with baseline algorithms.

## II. THE SEQUENTIAL RANK-ONE CONSTRAINT RELAXATION ALGORITHM

Inspired by existing rank-one optimization methods, in this section we propose a novel *sequential rank-one constraint relaxation* algorithm to generate a guaranteed rank-one solution, which can be shown to be a Karush-Kuhn-Tucker (KKT) stationary solution to the original optimization problem (1).

### A. Rank-One Constraint Reformulation and Analysis

Considering that  $\text{rank}(\mathbf{X})$  is a discontinuous non-convex function and does not have a closed-form expression [2], we reformulate  $\text{rank}(\mathbf{X}) = 1$  to the following equivalent form:

$$\lambda_{\max}(\mathbf{X}) = \text{Tr}(\mathbf{X}), \quad (3)$$

where  $\lambda_{\max}(\mathbf{X})$  and  $\text{Tr}(\mathbf{X})$  denote the largest eigenvalue and trace of  $\mathbf{X}$ , respectively. Now, (3) involves continuous functions of  $\mathbf{X}$  but still a non-convex constraint. As we know that  $\lambda_{\max}(\mathbf{X})$  can be equivalently represented as

$$\lambda_{\max}(\mathbf{X}) = \max_{\mathbf{v} \in \mathbb{C}^{N \times 1}, \|\mathbf{v}\|_2 \leq 1} \mathbf{v}^H \mathbf{X} \mathbf{v}, \quad (4)$$

(3) can be further formulated as

$$\max_{\mathbf{v} \in \mathbb{C}^{N \times 1}, \|\mathbf{v}\|_2 \leq 1} \mathbf{v}^H \mathbf{X} \mathbf{v} = \text{Tr}(\mathbf{X}). \quad (5)$$

### B. Alternating Optimization Algorithm

Based on the equivalent rank-one constraint (5), we can consider the vector  $\mathbf{v}$  as a slack variable for the original problem (1). Then, Problem (1) is equivalent to the following joint optimization problem with respect to  $\{\mathbf{v}, \mathbf{X}\}$ :

$$\min_{\mathbf{X} \succeq \mathbf{0}, \mathbf{v}} g_0(\mathbf{X}) \quad (6a)$$

$$\text{s.t. } g_k(\mathbf{X}) \leq_k b_k, \quad k = 1, \dots, K \quad (6b)$$

$$\max_{\mathbf{v} \in \mathbb{C}^{N \times 1}, \|\mathbf{v}\|_2 \leq 1} \mathbf{v}^H \mathbf{X} \mathbf{v} = \text{Tr}(\mathbf{X}). \quad (6c)$$

Suppose that we find a *feasible* point  $\mathbf{X}^{(i)}$  for Problem (6). The constraint (6c) implies that  $\mathbf{X}^{(i)}$  must be rank-one to ensure  $\lambda_{\max}(\mathbf{X}^{(i)}) = \text{Tr}(\mathbf{X}^{(i)})$ . Thus, the optimization of Problem (6) with respect to  $\mathbf{v}$  yields an optimal solution

$$\mathbf{v}^{(i)} = \mathbf{u}_{\max}(\mathbf{X}^{(i)}). \quad (7)$$

On the other hand, for any given  $\mathbf{v}^{(i)}$  in (7), the non-convex constraint (6c) becomes a linear constraint

$$\mathbf{v}^{(i),H} \mathbf{X} \mathbf{v}^{(i)} = \text{Tr}(\mathbf{X}), \quad (8)$$

because (8) is a linear constraint on  $\mathbf{X}$ . Then, Problem (6) becomes a convex optimization problem with respect to the variable  $\mathbf{X}$ , and thus the optimal solution  $\mathbf{X}^{(i+1)}$  can be obtained by using standard convex optimization methods.

However, to make (8) hold, the solution  $\mathbf{X}^{(i+1)}$  has to be rank-one and its principal eigenvector should be equal to  $\mathbf{v}^{(i)}$ , i.e., the direction of  $\mathbf{u}_{\max}(\mathbf{X}^{(i+1)})$  should be parallel to that of  $\mathbf{u}_{\max}(\mathbf{X}^{(i)})$ . This alternating optimization of  $\{\mathbf{v}, \mathbf{X}\}$  cannot improve the solution quality compared with the initial feasible point. Furthermore, it is still unknown how to obtain a feasible solution for the generic problem (1), as the difficulty of finding a feasible solution for generic NP-hard optimization problems with multiple constraints is even comparable with that of obtaining its optimal solution [13].

### C. SROCR Optimization Idea and Algorithm

To break the bottleneck of the rank-one constraint, we propose a novel algorithm by *sequentially* relaxing the rank-one constraint rather than completely dropping it as is done in traditional algorithms.

1) *Idea of SROCR*: We propose to *partially* relax the rank one constraint  $\lambda_{\max}(\mathbf{X}) = \text{Tr}(\mathbf{X})$  to

$$\lambda_{\max}(\mathbf{X}) \geq w^{(i)} \text{Tr}(\mathbf{X}), \quad (9)$$

where  $w^{(i)} \in [0, 1]$  denotes a relaxation parameter that controls the largest eigenvalue to trace ratio of  $\mathbf{X}$ , as the solution  $\mathbf{X}^*$  based on (9) is guaranteed to satisfy the following condition:

$$w^{(i)} \leq \frac{\lambda_{\max}(\mathbf{X}^*)}{\text{Tr}(\mathbf{X}^*)} \leq 1. \quad (10)$$

Then, the constraint set of (9) gradually shrinks and  $\mathbf{X}^*$  can approach 1 as  $w^{(i)}$  increases. One significant benefit of the above partial rank-one constraint relaxation scheme is that it makes it easier to find a feasible solution  $\mathbf{X}^{(0)}$  for the optimization with a small  $w^{(i)}$ . For example, when  $w^{(0)} = 0$ , (9) is equivalent to dropping the rank-one constraint. In this case, we can easily find a feasible point  $\mathbf{X}^{(0)}$  by solving the convex relaxed problem (2). This motivates us to increase  $w^{(i)}$  sequentially from 0 in iterations such that the constraint set (9) can gradually approach the real rank-one constraint set when  $w^{(i)} = 1$ .

2) *SROCR Algorithm*: The proposed SROCR algorithm can be described as follows.

In Problem (12), the constraint (12c) is jointly derived from (7), (8) and (9). In (15),  $\delta^{(i)}$  denotes the step size for weight parameter updating. To make  $\mathbf{X}^{(i)}$  more likely to be feasible for Problem (12) in the next iteration, in general the step size  $\{\delta^{(i)}\}$  should be small. If the pre-defined step size  $\delta^{(i)}$  makes Problem (12) insoluble or infeasible, we can reduce the step size by (14) until Problem (12) is solvable.

**Algorithm 1** SROCR Algorithm for Problem (1)

Initialization:  $i = 0$ , convergence thresholds  $\epsilon_1, \epsilon_2$ .  
Solve the relaxed problem (2) and obtain  $\mathbf{X}^{(0)}$  when  $w^{(0)} = 0$ .  
Define an initial step size

$$\delta^{(0)} \in (0, 1 - \lambda_{\max}(\mathbf{X}^{(0)}) / \text{Tr}(\mathbf{X}^{(0)})]. \quad (11)$$

**repeat**

Given  $\{w^{(i)}, \mathbf{X}^{(i)}\}$ , solve the convex problem

$$\min_{\mathbf{X} \succeq \mathbf{0}} g_0(\mathbf{X}) \quad (12a)$$

$$\text{s.t. } g_k(\mathbf{X}) \leq b_k, \quad k = 1, \dots, K \quad (12b)$$

$$\mathbf{u}_{\max}(\mathbf{X}^{(i)})^H \mathbf{X} \mathbf{u}_{\max}(\mathbf{X}^{(i)}) \geq w^{(i)} \text{Tr}(\mathbf{X}). \quad (12c)$$

**if** Problem (12) is solvable **then**

Obtain the optimal solution  $\mathbf{X}^{(i+1)}$  for Problem (12);

$$\delta^{(i+1)} \leftarrow \delta^{(0)}; \quad (13)$$

**else**

$$\delta^{(i+1)} \leftarrow \delta^{(i)} / 2. \quad (14)$$

$$w^{(i+1)} \leftarrow \min \left( 1, \frac{\lambda_{\max}(\mathbf{X}^{(i+1)})}{\text{Tr}(\mathbf{X}^{(i+1)})} + \delta^{(i+1)} \right). \quad (15)$$

$i \leftarrow i + 1$ .

**until**  $w^{(i-1)} \geq \epsilon_1$  &  $|g_0(\mathbf{X}^{(i)}) - g_0(\mathbf{X}^{(i-1)})| \leq \epsilon_2$ ;

**Remark 1** The proposed SROCR Algorithm 1 makes the procedure of rank-one optimization easier to analyze. In principle, Algorithm 1 gradually projects the largest eigenvector of the matrix variable  $\mathbf{X}$  onto an updated direction  $\mathbf{v}^{(i)}$  until all the power lies in a one-dimensional subspace. In addition, unlike penalty based methods,  $\mathbf{X}^{(i)}$  approaches a rank-one solution by monotonically increasing  $\{w^{(i)}\}$  in an iterative but controllable manner.

The step size reduction in (14) could be extended to the general case  $\delta^{(i+1)} \leftarrow \delta^{(i)} / L$  where  $L \geq 2$  in order to speed up making Problem (12) solvable if it is infeasible in the last iteration. In the first iteration, the relaxation weight  $w^{(1)}$  computed based on (15) satisfies  $1/N < w^{(1)} \leq 1$ , as the ratio of the largest eigenvalue to the trace of an arbitrary positive semidefinite matrix should be not less than  $1/N$ .  $\square$

**Theorem 1** Given a feasible initial point  $\mathbf{X}^{(0)}$ , Algorithm 1 converges to a KKT stationary solution of Problem (6), equivalent to Problem (1).  $\square$

*Proof:* Solving Problem (12) can be considered as an alternating optimization of  $\mathbf{v} = \mathbf{u}_{\max}(\mathbf{X})$  and  $\mathbf{X}$ . The proof is detailed in [14].  $\blacksquare$

**Remark 2** The performance of Algorithm 1 depends on the initialization. A large initial step size  $\delta^{(0)}$  in (11) may cause some infeasible iterations, but a small initial step size may require more iterations to increase  $w^{(0)}$  to 1. In addition, as Algorithm 1 is a projection procedure from an infeasible

solution to a feasible one, it may skip some good solutions in the projection procedure if the step size is too large. In contrast, a small step size usually leads to a smooth projection procedure, thereby yielding a better solution with a high probability. Therefore, we can easily determine an initial step size  $\delta^{(0)}$  according to the priority of complexity vs. accuracy.  $\square$

### III. NUMERICAL EXAMPLE

To illustrate this algorithm we consider an example – a cooperative interference channel of two transmitter-receiver pairs, in which each transmitter causes interference to the other receiver in a spectrum sharing scenario. We assume that each node is equipped with  $N$  antennas and a single radio frequency (RF) chain. Each transmitter employs beamforming to minimize its interference leakage but subject to its own desired signal strength constraint. Let  $\mathbf{x} \in \mathbb{C}^{N \times 1}$ ,  $\mathbf{H}_0 \in \mathbb{C}^{N \times N}$  and  $\mathbf{H}_1 \in \mathbb{C}^{N \times N}$  denote the analog beamforming vector of a transmitter, the interfering channel matrix to the undesired receiver and the desired channel matrix to its own receiver, respectively. Then, we can formulate the following optimization problem:

$$\min_{\mathbf{x} \in \mathbb{C}^{N \times 1}} \mathbf{x}^H \mathbf{H}_0^H \mathbf{H}_0 \mathbf{x} \quad (16a)$$

$$\text{s.t. } \mathbf{x}^H \mathbf{H}_1^H \mathbf{H}_1 \mathbf{x} \geq \gamma_0, \quad (16b)$$

$$|\mathbf{x}(n)| = 1, n = 1, \dots, N. \quad (16c)$$

where  $\gamma_0$  in (16b) denotes the minimal required signal strength to guarantee the quality of service (QoS) and is defined as  $\gamma_0 = \text{Tr}(\mathbf{H}_1^H \mathbf{H}_1) / N$ . Based on the SDP technique and the definition of  $\mathbf{X} \triangleq \mathbf{x} \mathbf{x}^H$ , Problem (16) is equivalent to

$$\min_{\mathbf{X} \succeq \mathbf{0}} \text{Tr}(\mathbf{H}_0^H \mathbf{H}_0 \mathbf{X}) \quad (17a)$$

$$\text{s.t. } \text{Tr}(\mathbf{H}_1^H \mathbf{H}_1 \mathbf{X}) \geq \gamma_0, \quad (17b)$$

$$\text{Tr}(\mathbf{E}_n \mathbf{X}) = 1, n = 1, \dots, N \quad (17c)$$

$$\text{rank}(\mathbf{X}) = 1, \quad (17d)$$

where  $\mathbf{E}_n$  is a zero  $N \times N$  matrix except for the  $n$ -th diagonal element being one. We assume Rayleigh fading channels in which the elements are independent and identically distributed with zero mean and unit variance. In addition, 100 random channels  $\{\mathbf{H}_0, \mathbf{H}_1\}$  for Problem (17) are evaluated for Monte Carlo simulations.

Let  $\mathbf{X}^{(0)}$  denote the solution to the relaxed problem (17a)-(17c). When  $N = 16$ , Fig. 1 shows that all the initial solutions  $\{\mathbf{X}^{(0)}\}$  for 100 random realizations of  $\{\mathbf{H}_0, \mathbf{H}_1\}$  are not rank-one. To extract a rank-one solution, the proposed SROCR algorithm is first evaluated. Fig. 2 and Fig. 3 show its convergence for one channel realization but with different initial step sizes<sup>1</sup>. In Fig. 2, we observe that the objective values converge gradually to different values at different speeds. Fig. 3 shows the convergence of the largest eigenvalue to trace ratio towards

<sup>1</sup>The algorithm iterates until  $\lambda_1(\mathbf{X}^{(i)}) / \text{Tr}(\mathbf{X}^{(i)}) \geq 0.999$  and  $|\text{Tr}(\mathbf{H}_0^H \mathbf{H}_0 \mathbf{X}^{(i)}) - \text{Tr}(\mathbf{H}_0^H \mathbf{H}_0 \mathbf{X}^{(i-1)})| \leq 10^{-4}$  are simultaneously satisfied. In addition, the maximal number of iterations is defined as 500. The convergence criterion for the algorithms used as baselines is similarly defined.

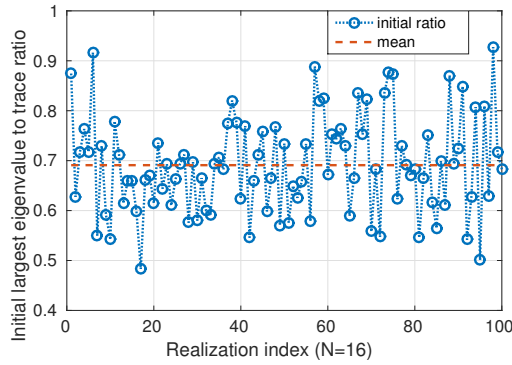


Fig. 1:  $\lambda_1(\mathbf{X}^{(0)})/\text{Tr}(\mathbf{X}^{(0)})$  for 100 channels when  $N = 16$ .

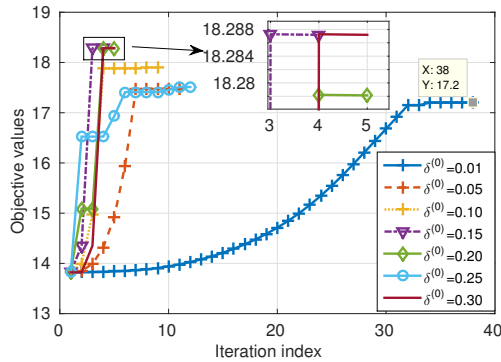


Fig. 2: Objective convergence via the SROCR algorithm for one channel realization with different values of the initial step size

1, which indicates a rank-one solution, for different values of  $\delta^0$ . The step size updating happens when the objective values or the largest eigenvalue to trace ratio stay constant in Fig. 2 and Fig. 3. This example implies that the proposed algorithm has a reasonably low complexity (within only a few tens of iterations) in general, and a small initial step size may lead to a better solution but at the cost of more iterations.

To illustrate the relationship between convergence and the initial step size, Figs. 4-5 show average convergence results for 100 random channel realizations versus the initial step size. Interestingly, when the initial step size decreases, Fig. 4 shows that the average converged objective value of the minimization problem (17) decreases as well, but Fig. 5 shows that the corresponding average complexity increases. This result is consistent with our assertion in Remark 1 for Algorithm 1 that a smaller step size enables smoother projections, which provides a high probability of achieving a better solution. Therefore, there exists a trade-off between performance and complexity. The converged objective values change slightly when  $\delta^{(0)} \leq 0.1$  in Fig. 4 and the convergence rate increases slightly when  $\delta^{(0)} \geq 0.1$  in Fig. 5. Thus,  $\delta^{(0)} = 0.1$  might be a good choice for this problem to achieve reasonable performance with low complexity. On the other hand, if the system has a high requirement in terms of objective performance,

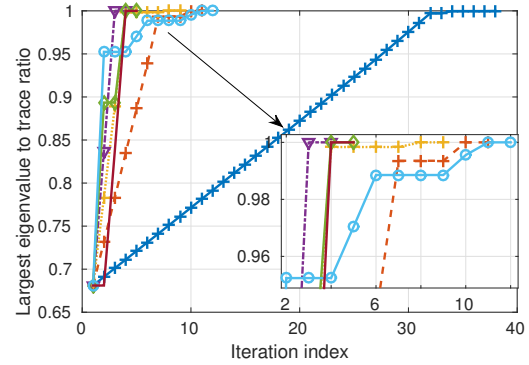


Fig. 3:  $\lambda_1(\mathbf{X}^{(i)})/\text{Tr}(\mathbf{X}^{(i)})$  convergence to 1 via the SROCR algorithm for one channel realization with different values of the initial step size (same legend as Fig. 2)

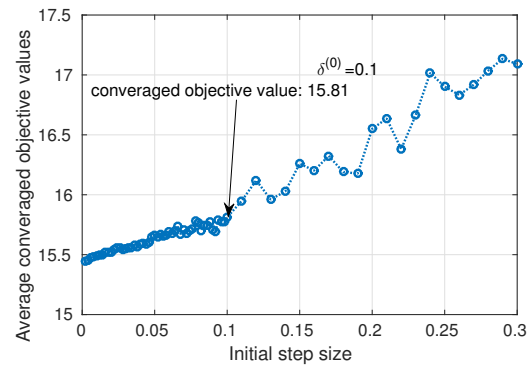


Fig. 4: Average objective value achieved by the SROCR algorithm vs. the initial step size

a smaller step size could be selected, e.g.,  $\delta^{(0)} = 0.05$  or  $\delta^{(0)} = 0.01$ .

Based on the same channel realizations, two recently proposed methods are evaluated as baselines for performance comparison, since both have already been shown to outperform most other related methods: 1) PSCA – the penalty based successive convex approximation proposed in [15] with an additional updating of the penalty parameter  $\mu^{(i)}$ ; and 2) FPP – the feasible point pursuit (FPP) algorithm in [11] with a fixed penalty parameter  $\mu^{(0)}$ . We summarize a performance comparison of the proposed algorithm with these two baseline algorithms with different numbers of antennas as shown in Table I. In particular, we run the proposed SROCR algorithm for  $\delta^{(0)} = 0.01$  and  $\delta^{(0)} = 0.1$ , and PSCA and FPP algorithms for  $\mu^{(0)} = 10$  and  $\mu^{(0)} = 1000$ , respectively. Table I shows that the proposed algorithm with  $\delta^{(0)} = 0.01$  can always converge to feasible solutions yielding smaller objective values compared with the baseline algorithms. Note also that PSCA usually outperforms FPP in terms of both objective values and convergence rate. Unlike the proposed algorithm, PSCA and FPP with some initial penalty parameters may lead to infeasible solutions, where the infeasibility is defined as either of the situations of convergence to an infeasible solution or non-

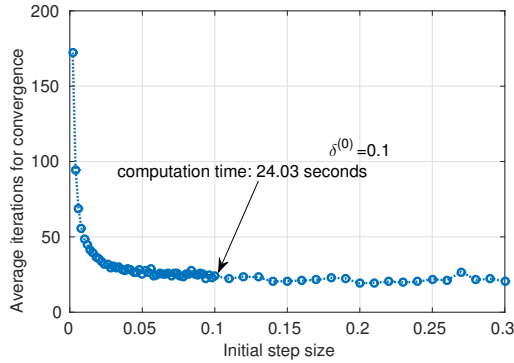


Fig. 5: Average number of iterations for convergence of the SROCR algorithm vs. initial step size

TABLE I: Average performance comparison: for each  $N$ , the first, second and third subrows denote converged real objective values (excluding the values of penalty terms), computation time [second] and probability of infeasibility.

$N$	Proposed	PSCA	FPP
	$\delta^{(0)}:0.01/0.1$	$\mu^{(0)}:10/1000$	$\mu^{(0)}:10/1000$
16	15.49/15.81	15.57/15.73	15.84/15.75
	15/9.82	6.71/21.82	27.76 /109.82
	0/0	0/0	0/2%
24	28.11/29.40	28.69/29.26	28.38/29.08
	21.87/12.03	9.55/42.39	75.67/194.27
	0/0	0/0	5%/6%
32	45.63/49.03	46.82/48.42	47.20/48.05
	40.83/11.85	17.44/63.25	232.57/287.49
	0/0	0/0	27%/6%
40	66.87/72.71	68.73/70.63	67.68/70.86
	58.32/18.13	22.75/114.09	442.45/347.05
	0/0	0/0	5%/4%
48	90.37/99.57	93.02/97.68	91.47/97.45
	82.90/22.09	30.54/151.07	769.78/447.56
	0/0	0/0	85%/9%
56	118.53/129.34	123.41/127.06	117.13/128.69
	79.97/27.61	41.48/234.93	984.57/545.59
	0/0	0/2%	99%/18%
64	149.09/166.62	153.32/160.60	144.72/163.43
	138.02/33.6	57.6/350.9	1110.1/598.9
	0/0	0/3%	100%/11%

convergence within 500 iterations. For example,  $\mu^{(0)} = 10$  usually results in an infeasible solution by FPP for large  $N$ . In addition, the same  $\mu^{(0)}$  for FPP and PSCA can result in feasible solutions for small  $N$  but not for large  $N$ . Therefore, it is in general unclear how to select good penalty parameters for the baseline algorithms. In contrast, we observe that the same  $\delta^{(0)}$  works robustly for the proposed algorithm with different

values of  $N$ .

#### IV. CONCLUSIONS

In this paper, we have considered rank-one constrained optimization problems. We have proposed the SROCR algorithm which achieves a locally optimal solution for a generic rank-one constrained problem by sequentially relaxing the rank-one constraint. We have seen that the proposed algorithm usually achieves a better final objective value but with a comparable or even lower complexity compared with baseline algorithms. Moreover, we have seen that the proposed algorithm works for generic rank-one constrained optimization problems, while, for example, the feasible point pursuit algorithm is only valid for quadratic programs.

#### REFERENCES

- [1] Z.-Q. Luo, W.-K. Ma, A.M.-C. So, Y. Ye, and S. Zhang, "Semidefinite relaxation of quadratic optimization problems," *IEEE Signal Process. Mag.*, vol. 27, no. 3, pp. 20–34, May 2010.
- [2] B. Recht, M. Fazel, and P. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM Rev.*, vol. 52, no. 3, pp. 471–501, 2010.
- [3] W. Ai, Y. Huang, and S. Zhang, "New results on Hermitian matrix rank-one decomposition," *Math. Program: Ser. A*, vol. 128, no. 1-2, pp. 253–283, Jun. 2011.
- [4] E. Song, Q. Shi, M. Sanjabi, R.-Y. Sun, and Z.-Q. Luo, "Robust SINR-constrained MISO downlink beamforming: when is semidefinite programming relaxation tight?," *EURASIP J. Wirel. Commun. Netw.*, pp. 1–11, Aug. 2012.
- [5] P. Tseng, "Further results on approximating nonconvex quadratic optimization by semidefinite programming relaxation," *SIAM J. Optim.*, vol. 14, no. 1, pp. 268–283, Jul. 2003.
- [6] S. A. Vorobyov, A. B. Gershman, and Z.-Q. Luo, "Robust adaptive beamforming using worst-case performance optimization: a solution to the signal mismatch problem," *IEEE Trans. Signal Process.*, vol. 51, no. 2, pp. 313–324, Feb. 2003.
- [7] N. D. Sidiropoulos, T. N. Davidson, and Z.-Q. Luo, "Transmit beamforming for physical-layer multicasting," *IEEE Trans. Signal Process.*, vol. 54, no. 6, pp. 2239–2251, Jun. 2006.
- [8] Z.-Q. Luo and T.-H. Chang, "SDP Relaxation of Homogeneous Quadratic Optimization: Approximation Bounds and Applications," in *Convex Optimization in Signal Processing and Communications*, D. P. Palomar and Y. C. Eldar, Eds. Cambridge Univ. Press, Cambridge, UK, 2010.
- [9] M. Fazel, H. Hindi, and S. Boyd, "A rank minimization heuristic with application to minimum order system approximation," in *Proc. American Control Conference*, Arlington, VA, Jun. 2001.
- [10] S. Burer and R. D. Monteiro, "A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization," *Math. Program.*, vol. 95, no. 2, pp. 329–357, Feb. 2003.
- [11] O. Mehanna, K. Huang, B. Gopalakrishnan, A. Konar, and N. Sidiropoulos, "Feasible point pursuit and successive approximation of non-convex QCQPs," *IEEE Signal Process. Lett.*, vol. 22, no. 7, pp. 804–808, 2015.
- [12] H. Yu and V. K. N. Lau, "Rank-constrained schur-convex optimization with multiple Trace/Log-Det constraints," *IEEE Trans. Signal Process.*, vol. 59, no. 1, pp. 304–314, Jan. 2011.
- [13] H. Tuy and N. Thuong, "On the global minimization of a convex function under general nonconvex constraints," *Applied Mathematics and Optimization*, vol. 18, no. 1, pp. 119–142, 1988.
- [14] P. Cao, J. Thompson, and H. V. Poor, "A sequential rank-one constraint relaxation algorithm and its applications in signal processing," preprint, 2017.
- [15] Y. Gao and D. Sun, "A majorized penalty approach for calibrating rank constrained a majorized penalty approach for calibrating rank constrained correlation matrix problems," Available online: <http://www.math.nus.edu.sg/matsundf/MajorPen.pdf>, Mar. 2010.