Quest Journals
Journal of Research in Applied Mathematics
Volume 9 ~ Issue 10 (2023) pp: 40-49
ISSN (Online): 2394-0743 ISSN (Print): 2394-0735
www.questjournals.org

Review Paper

On the Solution of the Pole Assignment Problem for Positive Systems

Fatma F.S. Omar*
*(Department of Mathematics, Faculty of Science/Gharyan University, Gharyan, Libya)
Corresponding Author:*

ABSTRACT: *Derivative-free optimization techniques are widely used for solving optimization problems. The focus in this work is given to some variants of Nelder-Mead and particle swarm methods to tackle two unconstrained optimization problems originated from optimal control, namely the pole assignment problem for discrete and continuous-time positive systems. we present the Nelder-Mead and Particle Swarm optimization methods to solve problems. Moreover comparing our results with benchmarks in the literature.*
KEYWORDS: *the pole assignment problem, positive systems, output feedback control, Nelder-Mead method, Particle Swarm Optimization.*

Received 04 Oct., 2023; Revised 25 Oct., 2023; Accepted 27 Oct., 2023 © The author(s) 2023. Published with open access at ***www.questjournals.org***

I. INTRODUCTION

The pole assignment problem (PAP) has been of great importance in many application areas. The problem in its simplest form was first addressed by Wonham [1] in 1967. Since then, huge number of publications have been proposed to tackle this problem. In 1976 Moore [2] identified the freedom available in pole assignment for the case of distinct eigenvalues. One of the first to address PAP by output feedback was Davison [3] and was extended by Davison and Chatterjee [4] and Sridhar and Lindhor [5]. Optimization techniques have been used for pole assignment via output feedback as demonstrated in Sobel and Shapiro [6] formulated an objective function to minimize the sum of the squares of the eigenvalue condition numbers subject to exact pole assignment. Mostafa et al. [7, 8, 9] tackled the PAP by gradient-based optimization methods. In optimal control literature there are various forms of the PAP for discrete and continuous–time systems that attract a lot of modern research among them [10-13].

Derivative-free methods have been appear since the 1950s (see [14]), where derivative-free methods were based on simplicity. Attractive thinking is often drawn from examples drawn in two dimensions. Has been a techniques used in scientific and engineering fields where it is best used to avoid calculating gradients. Derivative–free methods remain an effective option for several types of difficult optimization problems.

The importance of the research lies in the use of numerical methods in the solution that do not depend on the use of the derivative to obtain the stability state of the system as well as obtaining an initial point to be used in solving static output feedback design problems [9] .

This article is organized as follows. The next section introduces the formulations of the PAP problem for discrete and continuous–time systems together. In Section 3 we give some basic definitions. In Section 4 we give a brief description Algorithms Nelder-Mead (NM) simplex method and Particle Swarm Optimization (PSO) method for tackling the considered two problems. In Section 5 we demonstrate the performance of the methods on several test examples. Then we end with a conclusion.

Notations: The eigenvalues of a matrix $M \in R_{m\times m}$ are denoted by $\lambda_i(M)), i = 1, \ldots, m$. The Greek letter $\rho(M)$ denotes the spectral radius of a square matrix M .Sometimes and for the sake of simplicity we skip the arguments of the considered functions, e.g., we use v to denote $vec(K)$ that stretches a matrix K into long column vector v.

II. PROBLEM FORMULATION

Consider the linear time-invariant control system with the following state space realization

$$\begin{cases} \delta x = Ax + Bu, \\ y = Cx \end{cases} \qquad (1)$$

where δ is an operator indicating the time derivative d/dt for continuous-time systems and a forward unit time shift for discrete-time systems. The vectors $x \in \mathbb{R}^m$, $u \in \mathbb{R}^p$, and $y \in \mathbb{R}^r$ are the state, the control input, and the measured output vectors, respectively.

Moreover, $A \in \mathbb{R}_{m\times m}$, $B \in \mathbb{R}_{m\times p}$and $C \in \mathbb{R}_{r\times m}$ are given constant matrices. The control law $u = Ky$ is often used to close the above control system yielding

$$\delta x = (A + BKC)x := A(K)x, \qquad (2)$$

where $A(K) = A + BKC$ is the closed-loop system matrix and $K \in \mathbb{R}_{p\times r}$ is the output feedback gain matrix which represents the unknown.

The pole assignment problem is to find an output feedback gain matrix K provided that the closed-loop system is in satisfactory stage by shifting controllable eigenvalues to desirable locations in the complex plane. In particular, for discrete-time systems the PAP requires the spectral radius of the closed-loop system matrix $A(K)$ to be strictly within the unit circle in the complex plane. In other words, let A, B and C be given constant matrices. The PAP for discrete–time systems is to find $K \in R_{p\times r}$ that solves the following optimization problem:

$$\min f(K) = \rho(A + BKC)$$
$$s.t\ g(K) = \rho(A + BKC) - (1 - \tau) \leq 0, \quad (3)$$

Where $\tau \in (0, 1)$ is a given constant and $f: \mathbb{R}_{p\times r} \to \mathbb{R}^+$ is generally nonconvex and non-smooth function and $\rho(\cdot)$ is the spectral radius of the matrix function $A + BKC$.

The inequality constraint of the problem (3) represents a cut to the objective function, where the major task is to find a feasible point K to the problem (3). In fact this problem can be regarded as the following unconstrained minimization problem:

$$\min f(K) = \rho(A + BKC) \qquad (4)$$

while we make sure of fulfilling the associated inequality constraint during the iterations of the proposed method. Obviously we can stop the unconstrained minimization solver as soon as the objective function becomes strictly less than one. However, we can run the method until it achieves a satisfactory stability margin as indicated by the inequality constraint. From a computational point of view, the unconstrained minimization solver when applied to the problem (4) it typically reduces the objective function where the imposed cut is fulfilled after finite number of iterations. Therefore, the focus is given to approximately solve the unconstrained minimization problem (4). On the other hand, the PAP for continuous–time control systems can be stated as follows. Let $A, B,$ and C be given constant matrices and let $\tilde{\lambda}_1, \tilde{\lambda}_2, \tilde{\lambda}_3, \cdots, \tilde{\lambda}_m \in \mathbb{C}$, be given desired eigenvalues. The PAP is to find a matrix variable $K \in \mathbb{R}_{p\times r}$ such that

$$\lambda_i(A + BKC) = \tilde{\lambda}_i, \qquad i = 1, 2, \ldots, m.$$

The PAP for continuous–time systems can be equivalently rewritten as: Find $K \in R_{p\times r}$ that solves the following nonlinear least-squares problem; see [7, 18]

$$\min f^{\hat{}}(K) = \frac{1}{2}\sum_{i=1}^{m}\left(\lambda_i\left(A(K) - \tilde{\lambda}_i\right)\right) * \left(\lambda_i\left(A(K) - \tilde{\lambda}_i\right)\right) \quad (5)$$

where the superscript $*$ denotes the complex conjugate.

Let us consider in the following some basic definitions of linear positive systems.

III. BASIC DEFINITIONS OF LINER POSITIVE SYSTEMS

Positive systems represent a class of control systems that have the property that its state variables are never negative for a given positive initial state. These systems appear in practical applications such as communication networks [15] and medical engineering [17] as these variables represent nonnegative physical quantities e.g. levels, heights, concentrations, etc. It is well known that positivity imposes a specific sign pattern on the system matrix A.

In particular, a discrete-time positive system of form (1) is fully characterized by having a positive matrix A while a continuous-time positive system is fully characterized by having a Metzler matrix A. Hence, we will concentrate on the eigenvalue regions of those kind of matrices only, see [17] and [19] for details.

Definition 1 [17] The linear control system (1) for the discrete-time case is said to be positive if for any nonnegative initial condition $x_0 \in \mathbb{R}_+^m$ and $u(t) \in \mathbb{R}_+^p$, the corresponding trajectory $x(t) \in \mathbb{R}_+^m$ for all $t \geq 0$.
We refer to [19] for the proof of the following theorems.

Theorem 1 A discrete-time linear system (1) is positive if and only if A, B and C have nonnegative elements, i.e., $a_{ij} \geq 0, b_{ij} \geq 0, c_{ij} \geq 0$ for all i, j.

Lemma 1 The system (1) for the continuous-time case is positive if and only if A has nonnegative off-diagonal elements, i.e., $a_{ij} \geq 0, i \neq j$, and B has nonnegative elements, i.e., $B_{ij} \geq 0$ for all i, j.

Definition 2 A square real matrix A is called a Metzler matrix if its off-diagonal elements are nonnegative, i.e., $a_{ij} \geq 0, i \neq j$.
Consequence of definition 2 is that A is Metzler matrix if and only if there exists $\beta \geq 0$ such that $A + \beta \times I_m \in \mathbb{R}_{m\times m}$.

Remark 1 The system (1) for the continuous-time case is positive if and only if A is Metzler matrix and B , C have nonnegative elements.

Definition 3 The linear positive system (1) is said to be positively stabilizable if for all $x_0 \geq 0$ there exists an input $u(t)$ such that for all $t \geq 0$ the state trajectories $x(t)$ is such that $x(t) \geq 0$ and $x(t) \to 0$ as t → ∞.

Definition 4 The spectral radius of a matrix $A \in \mathbb{C}_{m\times m}$ with eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_m$ is defined as $\rho(A) = \max\{|\lambda_i| : i \in \{1,2,\ldots,m\}\}$.

Definition 5 The spectral abscissa of matrix $A \in \mathbb{C}_{m\times m}$ with the set of all eigenvalues $\sigma(A)$ is defined as $\{Re(\lambda_i) : \lambda \in \sigma(A)\}$.

$$\mu(A) = \max_{i \in \{1,\ldots,m\}} \{Re(\lambda_i) : \lambda_i \in \sigma(A)\}.$$

Definition 6 For the continuous-time case, the control system (1) is asymptotically stable if and only if $\mu(A(K)) < 0$ while the asymptotically stability for the discrete-time case requires $\rho(A(K)) < 0$.

In this work we will apply the Nelder-Mead (NM) simplex method [20], and The particle swarm optimization (PSO) method [21] to tackle the two problems (4) and (5) in the case of the positive systems.

IV. NELDER-MEAD SIMPLEX AND PARTICLE SWARM OPTIMIZATION METHODS FOR PAP

In papers [22] and [23] we explained in detail the Nelder Mead (NM) simplex method as well as the The particle swarm optimization (PSO) method to solve the PAP in discrete and continuous-time systems. Therefore, we will suffice with setting the algorithms of the PAP for positive discrete and continuous-time systems.
How to handle positivity for the discrete and continuous-time system within Algorithms 3.1 and 4.1 in [22]?
• Initially, we choose the matrices A, B, C, have nonnegative elements in the case discrete-time system and choose a matrix A is Metzler and B , C have nonnegative elements in the case continuous-time.
• Within main loop for discrete-time system we require the corresponding state vector be such that

$$x_j(t) = A(K_{i+k+1})^t x0 \geq 0,$$

$$\forall t > 0, j = 1, \ldots, m, i = 0, \ldots, n$$

Also in the main loop for continuous-time system we require the corresponding state vector be such that

$$x_j(t) = e^{tA(K_{i,0})x_0} \geq 0,$$

$$\forall t > 0, j = 1, \ldots, m, i = 0, \ldots, n.$$

To tackle problems (4) and (5) we will use the same Algorithms 3.1 and 4.1 in the paper [22] with a simple change in begin, as well as adding a condition to the main loop in the two Algorithms ensures the positive.

Algorithm 1 (Nelder-Mead method for solving Problem (4))

1. Let A, B, C be given constant matrices have nonnegative elements, and let $\alpha > 0, \beta > 1, 0 < \gamma < 1$ and $0 < \delta < 1$ be given constants. Choose $K_0 \in \mathbb{R}_{p\times r}$ and $x_0 \in \mathbb{R}_{m\times 1}$ such that $x_0 \geq 0$. The vector $v_0 \in \mathbb{R}_n$ such that $v_0 = vec(K_0)$ to be one of the initial simplex vertices, the matrices $A(K_{i,0})$ have nonnegative elements and compute $f(K_{i,0}), i = 0, \dots, n$. Arrange the $n+1$ vertices so that
$$f(K_{0,k}) \leq f(K_{1,k}) \leq \dots \leq f(K_{n,k}),$$
hold. Identify v_0^b, v_0^w and compute $f(K_0^b)$. If $f(K_0^b) < Tol\rho$, stop; otherwise set $k \leftarrow 0$ and go to next step.
2. While $f(K_k) \geq Tol\rho$ and $k < k_{MaxIte}$, do
   i. Compute $\bar{v}_{k+1}$, set $v_{k+1}^{re} = \bar{v}_{k+1} + \alpha(\bar{v}_{k+1} - v_{k+1}^{W})$, reshape v_{k+1}^{re} as K_{k+1}^{re} and compute $f_{k+1}^{re} = f(K_{k+1}^{re})$ such that $A(K_{k+1}^{re})$ has nonnegative elements.
   ii. (Reflection step) If $f(K_{k+1}^{b}) \leq f_{k+1}^{re} < f(K_{k+1}^{W})$, set $v_{k+1}^{W} := v_{k+1}^{re}$; and go to step (vii)
   iii. (Expansion step) If $K_{k+1}^{re} < f(K_{k+1}^{b})$ then compute $v_{k+1}^{e} = \bar{v}_{k+1} + \beta(\bar{v}_{k+1} - v_{k+1}^{W})$, reshape v_{k+1}^{e} as K_{k+1}^{e} such that $A(K_{k+1}^{e})$ has nonnegative elements and compute $f_{k+1}^{e} = f(K_{k+1}^{e})$. If $f_{k+1}^{e} < f_{k+1}^{re}$, set $v_{k+1}^{W} := v_{k+1}^{e}$, otherwise $v_{k+1}^{W} := v_{k+1}^{re}$ and go to step (vii)
   iv. (Outside contraction step) If $f(K_{i,k+1}) \leq f_{k+1}^{re} < f(K_{k+1}^{W}) \forall\, i = 0, \dots, n,\ i \neq w$, then compute $v_{k+1}^{oc} = \bar{v}_{k+1} + \gamma(v^{re} - \bar{v}_{k+1})$, reshape v_{k+1}^{oc} as K_{k+1}^{oc} such that $A(K_{k+1}^{oc})$ has nonnegative elements and compute $f_{k+1}^{oc} = f(K_{k+1}^{oc})$. If $f_{k+1}^{oc} \leq f(K_{k+1}^{re})$, set $v_{k+1}^{W} := v_{k+1}^{oc}$ and go to step (vii) otherwise go to step (vi)
   v. (Inside contraction step) If $f_{k+1}^{re} \geq f_{k+1}^{W}$, then compute $v_{k+1}^{ic} = \bar{v}_{k+1} - \gamma(v^{re} - \bar{v}_{k+1})$ reshape v_{k+1}^{ic} as K_{k+1}^{ic} such that $A(K_{k+1}^{ic})$ has nonnegative elements and compute $f_{k+1}^{ic} = f(K_{k+1}^{ic})$. If $f_{k+1}^{ic} \leq f(K_{k+1}^{W})$, set $v_{k+1}^{W} := v_{k+1}^{ic}$ and go to step (vii) otherwise go to step (vi).
   vi. (Shrinking step) Set $v_{i,k+1} = v_{k+1}^{b} + \delta(v_{i,k+1} - v_{k+1}^{b})$, reshape $v_{i,k+1}$ as $K_{i,k+1}$ such that $A(K_{i,k+1})$ has nonnegative elements and compute $f(K_{i,k+1})$ for all $i = 0, \dots, n, i \neq b$.
   vii. Arrange the $n+1$ vertices so that $f(K_{0,k}) \leq f(K_{1,k}) \leq \dots \leq f(K_{n,k})$, holds and identify v_{k+1}^{b} and v_{k+1}^{W}.
   viii. Reshape v_{k+1}^{b} as K_{k+1}^{b} and compute $f(K_{k+1}^{b})$. If $f(K_{k+1}^{b}) < Tol\rho$, stop; otherwise set $k \leftarrow k+1$ and go to step (i).

End (do)

Algorithm 2 (Nelder-Mead method for solving Problem (5))

1. Let A is Metzler matrix and B, C have nonnegative elements. Let $\alpha > 0, \beta > 1$, $0 < \gamma < 1$ and $0 < \delta < 1$ be given constants. Choose $K_0 \in \mathbb{R}^{p\times r}$ and $x_0 \in \mathbb{R}^{m\times 1}$ such that $x_0 \geq 0$. The vector $v_0 \in \mathbb{R}^n$ such that $v_0 = vec(K_0)$ to be one of the initial simplex vertices. Then generate the remaining n vertices such that $x_j(t) = e^{tA(K_{i,0})x_0} \geq 0, \forall t > 0, j = 1, \dots, m, i = 0, \dots, n$. Moreover choose the desired eigenvalue $\tilde{\lambda}_1, \tilde{\lambda}_2, \dots, \tilde{\lambda}_m \in \mathbb{C}$, compute $f^{\wedge}(K_{i,0}),\ i = 0, \dots, n$.
2. Arrange the $n+1$ vertices so that $f^{\wedge}(K_{0,k}) \leq f^{\wedge}(K_{1,k}) \leq \dots \leq f^{\wedge}(K_{n,k})$, holds and identify v_0^b and v_0^W
3. Reshape $v_{i,0}$ as $K_{i,0}, \forall\, i = 0, 1, \dots, n$ and compute the objective function $f^{\wedge}(K_{i,0})$. If $\sum_{i=0}^{n} f^{\wedge}(K_{i,0})/(n+1) \leq AverFun$, stop; otherwise set $k \leftarrow 0$ and go to next step.
4. While $\sum_{i=0}^{n} f^{\wedge}(K_i^k)/(n+1) \leq AverFun$ and $k < MaxIte$, do
   i. Calculate the same steps from (i) to (vii) in Algorithm 1 after substitution f with $f^{\wedge}$.
   ii. If $x_j(t) = e^{tA(K_{i,0})x_0} \geq 0, \forall t > 0, j = 1, \dots, m, i = 0, \dots, n$. stop; otherwise go to next step.
   iii. Reshape $v_{i,k+1}$ as $K_{i,k+1}, \forall i = 0, 1, \dots, n$ and compute the objective function $f^{\wedge}(K_{i,k+1}i)$. $\sum_{i=0}^{n} f^{\wedge}(K_{i,k+1})/(n+1) \leq AverFun$, stop; otherwise set $k \leftarrow k+1$ and go to (i).

End (do)

Algorithm 3 (Particle swarm optimization method for solving Problem (4))

1. (Initialization) Let A, B, C be given constant matrices have nonnegative elements, and let $\omega_{max} \in (0,1)$, $\omega_{min} \in (0,1)$, $c_1 \in (0,2)$ and $c_2 \in (0,2)$ be given constants. Choose $K_0 \in \mathbb{R}_{p\times r}$ and $x_0 \in \mathbb{R}_{m\times 1}$ such that $x_0 \geq 0$. The vector $v_0 \in \mathbb{R}_n$ such that $v_0 = vec(K_0)$ to be one of the initial simplex vertices. Then generate the remaining n vertices such that $x_j(t) = A(K_0^i)^t x_0 \geq 0,\ \forall t > 0, j = 1, \dots, m,\ i = 0, \dots, n$. Calculate fitness of particles $F_i(0) = f(K_i(0));\ i = 0, \dots, n$. and find the index of the best particle b. Select $P_i(0) = v_i(0)$; $\forall i$ and $g^b(0) = v^b(0)$.

2. Reshape $g^b(0)$ as $K_{g^b}(0)$ and compute the objective function $F_{g^b}(0) = f(K_{g^b}(0))$. If $F_{g^b}(0) < Tol\rho$, stop; otherwise set $k \leftarrow 0$ and go to next step.
3. While $F_{g^b}(0) \geq Tol\rho$ and $k < MaxIte$, do
   i. Compute $\omega(k)$ using $\omega(k) = \omega_{max} - \frac{k}{k_{max}}(\omega_{max} - \omega_{min})$.
   ii. Update the velocity $V_i(k+1)$ and the position $v_i(k+1)$ of particles using
   $V_{i,j}(k+1) = \omega V_{i,j}(k) + c_1 rand_1(v_{i,j}^b(k) - v_{i,j}(k)) + c_2 rand_2(g_j^b(k) - v_{i,j}(k)), \forall i,j,$
   and $v_{i,j}(k+1) = v_{i,j}(k) + V_{i,j}(k+1), \forall i,j$, respectively
   iii. Reshape the positions $v_i(k+1)$ into the matrices $K_i(k+1)$ and compute the fitness $F_i^{k+1} = f(K_i(k+1))$, $i = 0,1,\dots,n$ and identify the index of the best particle $v^{b1}(k+1)$.
   iv. Update $v^b(k+1)$ for all the population $i = 0,1,\dots,n$: If $F_i(k+1) < F_i(k)$ set $v_i^b(k+1) = v_i(k+1)$ else set $v_i^b(k+1) = v_i^b(k)$.
   v. Update $g^b(k+1)$ of the population:
   If $F_{b1}(k+1) < F_b(k)$ set $g^b(k+1) = v^{b1}(k+1)$ and $v^b(k+1) = v^{b1}(k+1)$ else $g^b(k+1) = g^b(k)$.
   vi. If $x_j(t) = A(K_0^i)^t x_0 \geq 0,\ \forall t > 0, j = 1,\dots,m,\ i = 0,\dots,n$. stop; otherwise go to next step.
   vii. Reshape $g^b(k+1)$ as the matrix $K_{g^b}(k+1)$ and compute the corresponding fitness function $F_{g^b}(k+1) = f\left(K_{g^b}(k+1)\right)$. If $F_{g^b}(k+1) < Tol\rho$, stop; otherwise set $k \leftarrow k+1$ and go to step (i)

   End (do)

Algorithm 4 (Particle swarm optimization method for solving Problem (5))

1. Let A is Metzler matrix and B, C have nonnegative elements, Let $\omega_{max} \in (0,1)$, $\omega_{min} \in (0,1)$, $c_1 \in (0,2)$ and $c_2 \in (0,2)$ be given constants. Choose $K_0 \in \mathbb{R}^{p\times r}$ and $x_0 \in \mathbb{R}^{m\times 1}$ such that $x_0 \geq 0$. The vector $v_0 \in \mathbb{R}^n$ such that $v_0 = vec(K_0)$ to be one of the initial simplex vertices and $x_j(t) = e^{(tA(K_i^0)x_0)} \geq 0, \forall t > 0, j = 1,\dots,m, i = 0,\dots,n$. Moreover choose the desired eigenvalue $\tilde{\lambda}_1, \tilde{\lambda}_2, \tilde{\lambda}_3, \cdots, \tilde{\lambda}_m \in \mathbb{C}$, calculate fitness of particles $F_i^0 = f^{\hat{}}(K_i^K)$; $\forall i$ and find the index of the best particle b. Select $P_i^0 = v_i^0$; $\forall i$ and $g^0 = v_b^0$
2. Reshape g^0 as K_{g^0} and compute the objective function $f^{\hat{}}(K_{g^0})$. If $\sum_{i=0}^n f^{\hat{}}(K_i^0)/(n+1) \leq AverFun$, stop; otherwise set $k \leftarrow 0$ and go to next step.
3. While $\sum_{i=0}^n f^{\hat{}}(K_i^0)/(n+1) \leq AverFun$ and $k < MaxIte$, do
   i. Calculate (i)–(v) in Algorithm 3 after change f to be replaced by the objective function $f^{\hat{}}$.
   ii. If $x_j(t) = e^{tA(K_{i,0})x_0} \geq 0, \forall t > 0, j = 1,\dots,m, i = 0,\dots,n$. stop; otherwise go to next step.
   iii. Reshape v_i^{k+1} as K_i^{k+1} and compute the objective function $f^{\hat{}}(K_i^{k+1})$.
   iv. If $\sum_{i=0}^n f^{\hat{}}(K_i^{k+1})/(n+1) \leq AverFun$, stop; otherwise set $k \leftarrow k+1$ and go to

End (do)

V. NUMERICAL RESULTS

In this section, various test problems are provided to illustrate the performance NM and PSO methods. The compared with respect to number of iterations and CPU time (sec.), the starting point is often generated randomly. Sometimes we start with the zero matrix or a matrix with ones on its entries. The problem dimensions are (m,p,r). The methods are implemented using MATLAB and all results are using a 1.80 Ghz Pentium 2 CPU with $2048MB\ RAM$. Some of the considered test problems are for continuous–time systems. The MATLAB function c2d from the control toolbox is employed to provide the corresponding discrete–time data matrices. In our experiment, we used the following values for termination:

$$\sigma^{\hat{}} = 0.5, \quad MaxIter = 500, \quad AverFun = 10^{-6},$$

$$Tol\rho = 0.98, \quad eps = 10^{-8}.$$

In the following we consider six examples to show the performance of the methods NM and PSO for the case of positive control systems.

Example 1 This test problem is borrowed from [6], the data matrices for the corresponding discrete-time system are the following

$$A = \begin{bmatrix} 0.4 & 0.6 \\ 0.6 & 0.4 \end{bmatrix}, B = \begin{bmatrix} 0.1 \\ 0.2 \end{bmatrix}, C^T = \begin{bmatrix} 0.2 \\ 0.5 \end{bmatrix}.$$

The spectral radius of the system matrix A is 1.0000. The objective function values at starting matrix K_0 are $f(A(K_0))^{NM} = 1.0707 = f(A(K_{init}))^{PSO}$, the methods NM and PSO stabilizing output feedback controllers after 2 and 4 iterations and CPU times 0.36 and 0.23 respectively. The corresponding objective function values at K_{fin} are $f(A(K_{fin}))^{NM} = 7.6368e-01$ and $f(A(K_{fin}))^{PSO} = 9.2222e-01$. The starting matrix and achieved output feedback matrices are the following

$$K_0 = [\ 0.6676\ ],$$
$$K_{fin}^{NM} = [-8.3125],\ K_{fin}^{PSO} = [\ -0.7481]$$

Example 2 This test problem is borrowed from [26], the data matrices for the corresponding discrete-time system are the following

$$A = \begin{bmatrix} 0.9 & 0.1 \\ 0.6 & 0.5 \end{bmatrix}, B = \begin{bmatrix} 0.9 \\ 0.8 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

The spectral radius of the system matrix A is 1.0162. The objective function values at starting matrix K_0 are $f(A(K_0))^{NM} = 1.9145 = f(A(K_{init}))^{PSO}$, the methods NM and PSO stabilizing output feedback controllers after 2 and 3 iterations and CPU times 0.44 and 0.27 respectively. The corresponding objective function values at K_{fin} are $f(A(K_{fin}))^{NM} = 7.5554e-01$ and $f(A(K_{fin}))^{PSO} = 7.7672e-01$. The starting matrix and achieved output feedback matrices are the following

$$K_0 = [1.0162 \quad 0.0000\,],$$
$$K_{fin}^{NM} = [\ -0.4838 \quad 0.1250],$$
$$K_{fin}^{PSO} = [-0.2666 \quad -0.0145].$$

Example 3 This test problem is borrowed from [6], the data matrices for the corresponding discrete-time system are the following

$$A = \begin{bmatrix} 0.6 & 0.6 \\ 0.6 & 0.4 \end{bmatrix}, B = \begin{bmatrix} 0.1 \\ 0.2 \end{bmatrix}, C^T = \begin{bmatrix} 0.2 \\ 0.5 \end{bmatrix}$$

The spectral radius of the system matrix A is 1.1083. The objective function values at starting matrix K_0 are $f(A(K_o))^{NM} = 1.1517 = f(A(K_{init}))^{PSO}$ the methods NM and PSO stabilizing output feedback controllers after 2 and 6 iterations and CPU times 0.33 and 0.27 respectively. The corresponding objective function values at K_{fin} are $f(A(K_{fin}))^{NM} = 8.7162e-01$ and $f(A(K_{fin}))^{PSO} = 9.5883e-01$. The starting matrix and achieved output feedback matrices are the following

$$K_0 = [\,0.4403],\ K_{fin}^{NM} = [-2.5597], \qquad K_{fin}^{PSO} = [\,-1.5781]$$

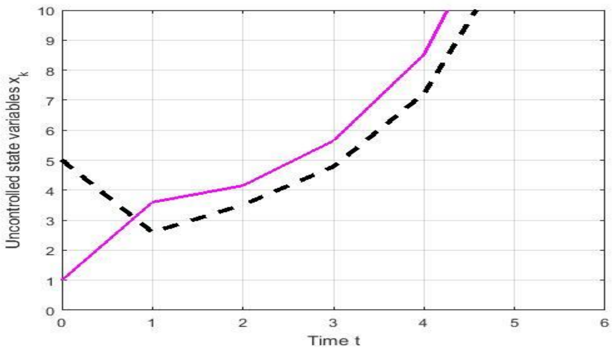

Figure 1: Uncontrolled state space variables for example 3

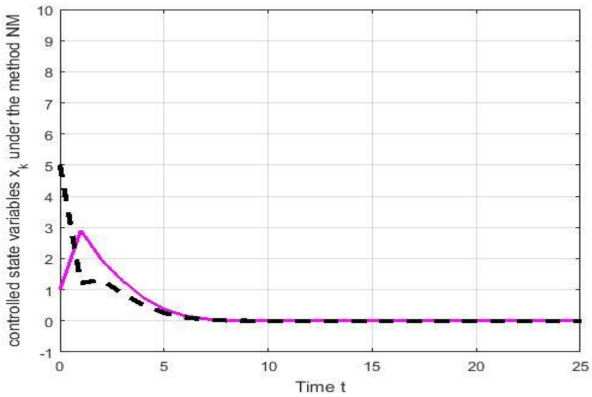

Figure 2: Controlled state space variables under the method NM for example 3

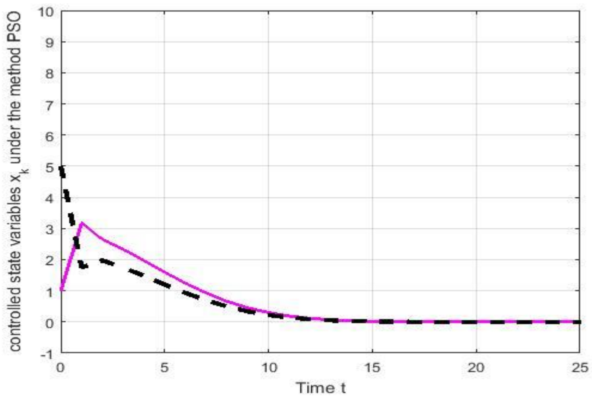

Figure 3: Controlled state space variables under the method PSO for example 3

Example 4 This test problem is borrowed from [24], which has the following data matrices are

$$A = \begin{bmatrix} -8 & 2 & 2 \\ 2 & -5 & 3 \\ 2 & 3 & -6 \end{bmatrix}, B = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$$

The system matrix A has the eigenvalues $(-9.2475, -8.4769, -1.2757)$. The desired eigenvalues $\tilde{\lambda}$ such that the shift is $s = 0.3$, Starting from the same K_0 where $f^{\hat{}}(K_0) = 1.7179$, the methods NM and PSO stabilizing output feedback controllers for the unconstrained problem (5) after 43 and 115 iterations and CPU times 0.73 and 0.45, respectively, to reach the least possible values of the objective function of the problem (5). The corresponding objective function values are

$$f^{\hat{}}(A(K_{fin}))^{NM} = 3.5379e-05,$$
$$f^{\hat{}}(A(K_{fin}))^{PSO} = 8.4099e-06.$$

The starting matrix K_0 and the output feedback gain matrices K_{fin}^{NM} and K_{fin}^{pso} are the following

$$K_0 = [1 \quad 1 \quad -2]^T$$
$$K_{fin}^{NM} = [2.9306 \quad 1.8657 \quad -0.8405]^T$$
$$K_{fin}^{PSO} = [\,2.9225 \quad 1.8605 \quad -0.8447]^T$$

Example 5 This test problem is borrowed from [25], which has the following data matrices are

$$A = \begin{bmatrix} -0.8 & 0.1 & 0.2 \\ 0.2 & -1.1 & 0.1 \\ 0.15 & 0.2 & -0.5 \end{bmatrix}, B = \begin{bmatrix} 0.1 \\ 0.5 \\ 0.4 \end{bmatrix}, C = \begin{bmatrix} 0.8 \\ 0.8 \\ 0.1 \end{bmatrix}$$

The system matrix A has the eigenvalues $(-0.3655, -0.8815, -1.1530)$. The desired eigenvalues $\tilde{\lambda}$ such that the shift is $s = 0.3$, Starting from the same K_0 where $f^{\hat{}}(K_0) = 2.6127$, the methods NM and PSO stabilizing output feedback controllers for the unconstrained problem (5) after 28 and 111 iterations and CPU times 0.62 and 0.27, respectively, to reach the least possible values of the objective function of the problem (5). The corresponding objective function values are

$$f\hat{}(A(K_{fin}))^{NM} = 7.3979e-03,$$
$$f\hat{}(A(K_{fin}))^{PSO} = 7.4158e-03.$$

The starting matrix K_0 and the output feedback gain matrices K_{fin}^{NM} and K_{fin}^{pso} are the following

$$K_0 = [\,2.0088]$$
$$K_{fin}^{NM} = [0.0944]$$
$$K_{fin}^{PSO} = [\,0.0890].$$

Example 6 This test problem is borrowed from [7], which has the following data matrices are

$$A = \begin{bmatrix} -1 & 1.1 \\ 1.1 & -1 \end{bmatrix}, B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

The system matrix A has the eigenvalues $(-2.1, 0.1)$. The desired eigenvalues $\tilde{\lambda}$ such that the shift is $s = 0.3$, Starting from the same K_0 where $f\hat{}(K_0) = 9.3021$, the methods NM and PSO stabilizing output feedback controllers for the unconstrained problem (5) after 46 and 11 iterations and CPU times 0.69 and 0.16, respectively, to reach the least possible values of the objective function of the problem (5). The corresponding objective function values are

$$f\hat{}(A(K_{fin}))^{NM} = 4.5789e-05,$$
$$f\hat{}(A(K_{fin}))^{PSO} = 6.3304e-06.$$

The starting matrix K_0 and the output feedback gain matrices K_{fin}^{NM} and K_{fin}^{pso} are the following

$$K_0 = \begin{bmatrix} 2.0000 & 2.0000 \\ 1.0000 & -0.0000 \end{bmatrix}$$
$$K_{fin}^{NM} = \begin{bmatrix} -0.5134 & 3.2687 \\ -0.8275 & -0.2785 \end{bmatrix}$$
$$K_{fin}^{PSO} = \begin{bmatrix} -0.8034 & 2.0000 \\ -0.7621 & 0.0000 \end{bmatrix}.$$

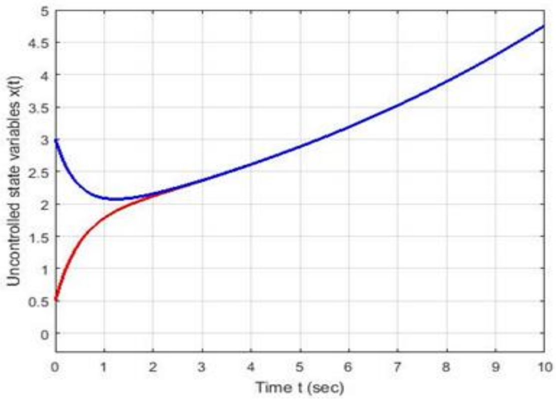

Figure 4: Uncontrolled state space variables for example 6

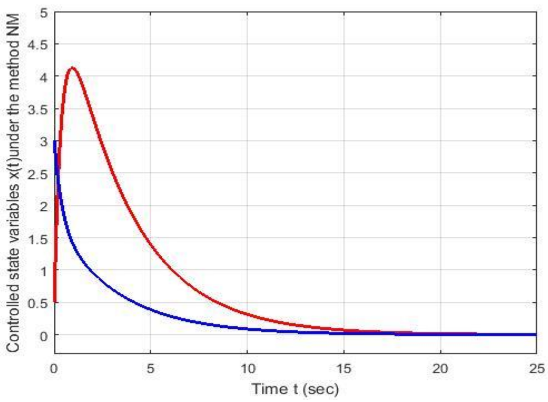

Figure 5: Controlled state space variables under the method NM for example 6

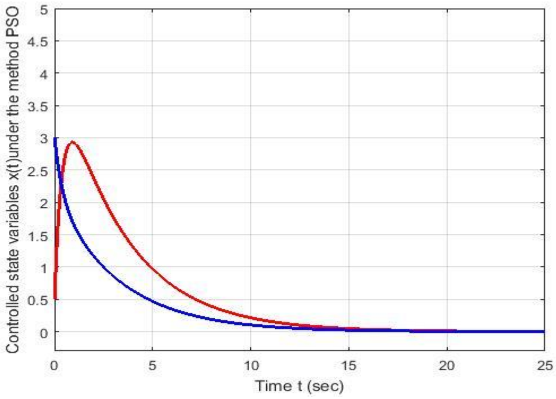

Figure 6: Controlled state space variables under the method PSO for example 6

VI. CONCLUSION

The pole assignment problem for discrete-time and continuous-time positive linear systems formulated as a semi-smooth unconstrained optimization, the derivative-free methods are proposed to find a local solution of the minimization problem or at least achieve a stabilizing output feedback gain matrix. The performance of the methods is demonstrated over wide range of test problems from the system and control literature. The two methods are successful in tackling the considered problems. In particular, the Nelder-Mead simplex method relatively outperforms the particle swarm optimization method.

REFERENCES

[1]. W. Wonham., (1967), On pole assignment in multi-input controllable linear systems, IEEE Transaction on Automatic Control, **12**:6 660–650.

[2]. B.C. Moore., (1976), On the flexibility offered by state feedback in multivariable systems beyond closed loop eigenvalue assignment, IEEE Trans, Automatic Control, **21**, 689–692.

[3]. E.J. Davison., (1970), On pole assignment in linear systems with incomplete state feedback, IEEE Transaction on Automatic Control, **15**, 348–351.

[4]. E.J. Davison and R. Chatterjee., (1971), A note on pole assignment in linear systems with incomplete state feedback, IEEE Trans, Automatic Control, AC, **16**, 98–99.

[5]. B. Sridhar and D. P. Lindorf, (1973), Pole placement with constraint gain output feedback, International Journal of Control, **18** 993–1003.

[6]. K.M. Sobel and E.Y. Shapiro., (1985), Design of decoupled longitudinal flight control laws utilizing eigensystem assignment, In Proceeding of the 1984 American Control Conference 403–408.

[7]. E.M.E. Mostafa, M.A. Tawhid and E.R. Elwan. (2014), Inexact Newton and quasi–Newton methods for the output feedback pole assignment problem, Computational and Applied Mathematics, **33** 517-542.

[8]. E.M.E. Mostafa, A.W. Aboutahoun and F.F.S. Omar., (2017), On the solution of the eigen- value assignment problem for discrete–time systems, Journal of Applied Mathematics, Vol., Article ID 7256769, **12** pages.

[9]. Fatma F.S. Omar., (May- June 2021), A Hybrid Conjugate Gradient Method for Discrete–Time Periodic Static Output Feedback Control Design, IOSR Journal of Mathematics (IOSR-JM)., e-ISSN: 2278-5728, p-ISSN: 2319-765X. **17**, 12-26

[10]. O. Bachelier, J. Bosche and D. Mehdi., (2006), On pole placement via eigenstructure assignment approach, IEEE Transactions on Automatic Control, **51**, 1554–1558

[11]. M. Fu. (2004), Pole placement via static output feedback is NP-hard, IEEE Transactions on Automatic Control, **49**, 855–857

[12]. R.E. Mahony and U. Helmke., (1995), System assignment and pole placement for symmetric realizations, Journal of Mathematical Systems, Estimation and Control, **5:2** 1–32

[13]. H.A. Tehrani and N. Ramroodi., (2013), Eigenvalue assignment of discrete–time linear systems with state and input time–delays, AUT Journal of Modeling and Simulation, **45**:2 23–30.

[14]. R.M. Lewis. V. Torczon and M.W. Trosset., (2000), Direct search methods, then and now, Journal of Computational and Applied Mathematics., **124**, 191–207.

[15]. X. Chen. Analysis and synthesis of positive systems under l_1 and L_1 performance. PhD. thesis, University of Hong Kong, 2017.

[16]. M. Colombino. Robust and Decentralized Control of Positive Systems: A Convex Approach, PhD. thesis, Imperial College of London, 2016.

[17]. L. Farina and S. Rinaldi. Positive linear systems, theory and applications, John Wiley & Sons, New York, 2000.

[18]. K. Yang and R. Orsi., (2007), Static output feedback pole assignment via a trust region approach, IEEE Transactions on Automatic Control, **52** 2146–2150.

[19]. D.G. Luenberger. Introduction to dynamic systems, theory, models and applications, John Wiley & Sons, New York, 1979.

[20]. J.A. Nelder and R. Mead., (1965), A simplex method for function minimization, Computer Journal , **7:4**, 308-313.

[21]. J. Kennedy, R.C. Eberhart., (1995), Particle swarm optimization, Proceedings of IEEE Conference on Neural Networks, **4** 1942-1948.

[22]. E.M.E. Mostafa and F.F.S. Omar., (2018), On the solution of the eigenvalue assignment Problem by derivative–free optimization methods, AMO–Advanced Modeling and Optimization, **30** 371–394

[23]. Fatma F.S. Omar., (2022), Inequality Constrained Optimization Technique for the Eigenvalue Assignment Problem., Gharyan University Journal, **23**., 229-243
[24]. V.T. Huynh and H. Trinh., (2017), Eigenvalue assignment for positive observers and its feasibility", European Journal of Control, **36** 10–17
[25]. Chen X, Lam J, Li P (2014) Positive filtering for continuous-time positive systems under L1 performance. Int J Control **87**(9):1906–1913
[26]. G. Rezaei, T. Binazadeh. B. Safarinejadian, Optimal Finite-time Control of Positive Linear Discrete-time systems