ICIC Express Letters ICIC International ©2018 ISSN 1881-803X
Volume **12**, Number **10**, October **2018** pp. **991–999**

ASSESSMENT OF IMPROVED P-TRANSFORMATION FOR GLOBAL OPTIMIZATION WITH APPLICATION ON DOUBLE INVERTED PENDULUM CONTROL

Maged Abdelnaby Kafafy, Mohamed Zohdy and Syed Faisal Haider

School of Engineering and Computer Science
Oakland University
318 Meadow Brook Rd., Rochester, MI 48309-4401, USA
{ makafafy; zohdyma }@oakland.edu; sfhaider1@yahoo.com

Received April 2018; accepted July 2018

Abstract. *In this paper, an improved P-transformation algorithm is proposed that utilizes the golden ratio as an improvement to solve global optimization. The proposed methodology was tested for several standard benchmark optimization problems as well as on optimizing the gain for a double inverted pendulum control. Our algorithm has shown promising results in terms of efficiency and accuracy when compared with the genetic algorithm.*

Keywords: Global optimization, Maximization, Minimization, P-transformation, Improved P-transformation, Continuous, Non-continuous functions, Golden ratio, Optimal control

1. **Introduction.** Finding the global optimum of the objective function is very important in engineering problems. Most existing optimization methods are essentially local in nature. Multi-dimensional, nonlinear objective functions have several extreme values and finding the global extremum of such functions is difficult. Chichinadze [1,4] has demonstrated that such problems can be solved by transforming the objective function through an Ψ transformation into a function $\Psi(\zeta)$ of one new variable ζ. He showed that the value of this transformed function decreases continuously to zero as the new variable is increased in value. As the value of the transformed function equals zero, the value of the new variable represents the global extremum of the original objective function. For simplicity, we will call this method the P-transformation method.

Adamczyk et al. [2] reformulated the P-transformation method using least squares parameter estimation and utilized the neural network for an optimal coordinate search. Later, Zohdy et al. [3] implemented a new global non-sequential search method for optimization in n-dimensions by utilizing a stochastic methodology for robust global optimization by extending the least squares parameter estimation. This work was therefore extended to both deterministic and probabilistic problems. They benchmarked their algorithm on a two-variable banana function. In this paper, we study the P-transformation algorithm as a universal heuristic optimization approach.

Our motivation derives from the desire to obtain global extremum for multi-dimensional objective functions, multi-objective functions, continuous, and non-continuous functions. Heuristic methods like the genetic algorithm [5-8] either fail to find the global extremum for such problems or are too expensive.

The main idea of the P-transformation is to convert the multi-dimensional objective function to an univariant objective function. In this paper, we propose an improvement to the existing P-transformation algorithm by utilizing the golden ratio [9]. We therefore applied the golden ratio to the transformed function, which accelerated the convergence

DOI: 10.24507/icicel.12.10.991

of the P-transformation method. Examples will demonstrate that our proposal is more efficient than the existing P-transformation and much more efficient than the genetic algorithm.

This paper is divided into five sections. In Section 2, we conduct a review of the P-transformation algorithm. In Section 3, we offer our proposed improvement to the P-transformation method. In Section 4, we first demonstrate the efficiency of our proposal by testing it on some benchmark problems and comparing the results with the P-transformation method. Then, we demonstrate the efficiency of our proposed method by comparing it with the genetic algorithm and the P-transformation method to optimize the controller gain for a double inverted pendulum problem. In Section 4, our findings are offered in Table 1 and Table 2 in 4.1 and 4.2 respectively. In Section 5, we conclude the paper.

2. **Review of the P-Transformation Algorithm.** In this section, we briefly describe the P-transformation algorithm as explained in the works of Adamczyk et al. [2]. The goal of the algorithm is to find the global maximum of an objective function. The transformation of the objective function, $f(x)$ where x is an n-dimensional vector, is carried out by a nonlinear operator by

$$P\{f(x)\} \rightarrow G(t) \tag{1}$$

where t is a scalar quantity that is taken to be pointwise Lebesgue's [10] division of the objective function $f(x)$

$$\min(f(x)) = t_0 < t_1 \cdots t_i < t_{\max} = \max(f(x)) \tag{2}$$

Let D be the set on which the function $f(x)$ is to be determined. D^* is a subset of D on which $f(x)$ also satisfies the constraints. Let D_i be the set on which $abs(f(x)) >= t_i$.

Given a function $H(x)$ such that

$$H(x) = \begin{cases} 0 & \text{at } D|D^* \\ 1 & \text{at } D^* \end{cases} \tag{3}$$

Riemann integral for the function above on the set D_i is equal to its measure μ_i in Equation (4) below.

$$\int_{D_i} \cdots \int H(x) dx_i \dots dx_n = \mu_i \tag{4}$$

The function $G(t)$ as illustrated in Figure 1 can be estimated from the integral shown in Equation (4). Set D_i can be computed using statistical tests. Let P_i be the probability that a point x randomly chosen from D^* also belongs to D_i and can be approximated by the ratio $s(r)/r$, where r is the number of randomly chosen points x (length of a, and b

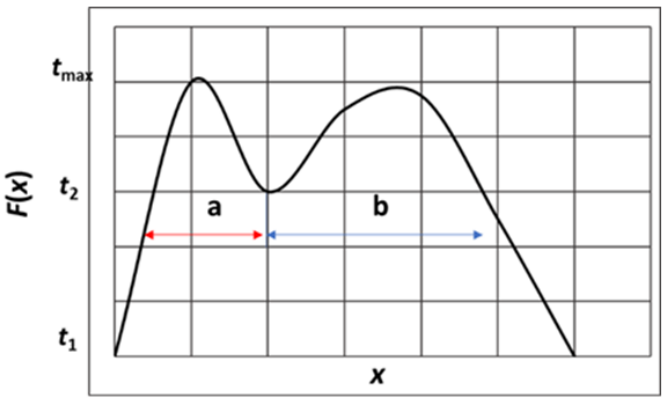

Figure 1. Type 1 – continuous objective function

on Figure 1) and $s(r)$ is the number of successes. Therefore, for any $\varepsilon > 0$, as $r \to \infty$, the ratio $s(r)/r$ can mimic the measure μ_i of the D_i set.

$$\lim_{\gamma \to \infty} \Pr\left(\frac{S(r)}{r} - P_i\right) > \varepsilon = 0 \tag{5}$$

$$\mu_i = s(r)/r \tag{6}$$

Once we have the μ_i's we can estimate the quadratic function $R(t)$ as shown in Equation (7) below

$$R(t) = a_0 t^2 + a_1 t + a_2 \tag{7}$$

where a_0, a_1, and a_2 can be found by utilizing the least squares approach with the following assumptions.

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \varphi^T \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{bmatrix} \tag{8}$$

where

$$\varphi = \begin{bmatrix} t_1^2 & t_1 & 1 \\ t_2^2 & t_2 & 1 \\ t_3^2 & t_3 & 1 \end{bmatrix} \tag{9}$$

The P-transformation algorithm [2,3] can be summarized as follows:

1- Generate N random vectors x_n
2- Check the constraints for each x_i
3- Evaluate the function $f(x_n)$ for all x_i's that satisfy the constraints
4- Let $t_1 = (\min f(x) + \max f(x))/2$
5- Calculate μ_i as defined in Equation (6)
6- Repeat step 5 for each t_i where $t_{i+1} = t_i + \Delta t$ until the condition in Equation (6) is no longer valid
7- Extrapolate $G(t)$ using $R(t)$ in Equation (7)
8- Find the roots for $R(t) = 0$ by calculating Equation (8)
   a. if $a_0 < 0$, then the global maximum

$$F(x^*) = \max(R_1, R_2),$$

   where R_1, R_2 are the roots of $R(t)$
   b. if $a_0 > 0$ then $F(x^*) = \min(Real(R_1, R_2))$
9- Repeat the algorithm until $\Delta F(x^*) << \varepsilon$.

The P-transformation also can be used on Type 2 and Type 3 functions as illustrated in Figure 2 and Figure 3 respectively.

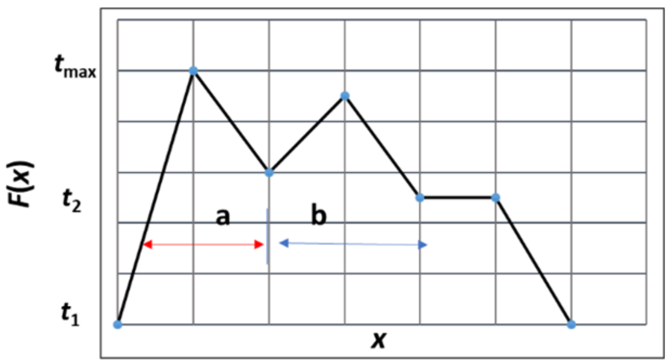

FIGURE 2. Type 2 – continuous objective function where its derivatives undergo breaks

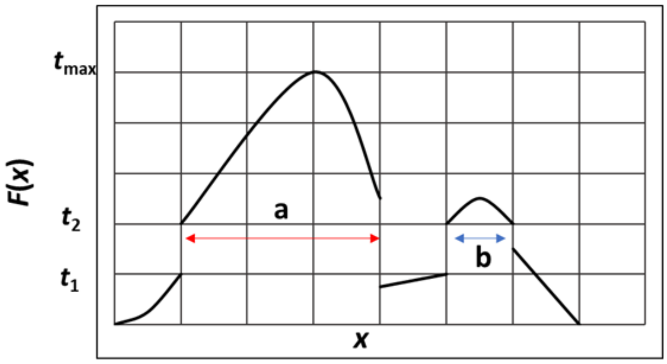

FIGURE 3. Type 3 – non-continuous objective function which is described by different equations

The P-transformation algorithm also can handle problems in the absence of the objective function or black box functions.

3. **Improved P-Transformation Algorithm.** Since the main idea of the P-transformation is to convert the multi-dimensional objective function to an univariant objective function, $G(t)$. This function has two significant distinguishing characters; first, $G(t)$ is monotonically decreasing as t increases in almost all types of functions, and second, the approximate value of the global extremum of the original objective function is determined as the value of t when $G(t) = 0$.

The golden section search is often used for finding the local optimum value for univariant models. We have applied the golden section search [18] to the $G(t)$ and found that first characteristic of the $G(t)$ steered the golden section search to search only in one direction. Therefore, we introduce the golden ratio (Fibonacci search) [9] to augment the search as an improvement to the existing algorithm.

The golden ratio is applied on $G(t)$ to calculate the new triplet of t's in steps (6) and (7) of the algorithm presented in the previous section. We calculated the triplet t's to follow the golden ratio pattern by the following equation:

$$\frac{t_{i+2} - t_i}{t_{i+1} + t_i} = \frac{t_{i+1} - t_i}{t_{i+2} + t_{i+1}} = \frac{1 + \sqrt{5}}{2} \tag{10}$$

We calculate the associated μ_i and apply the golden section search iteration on the triplets of t's points.

4. **Examples.** In this section, we first compare our method with the existing P-transformation method by testing it with common benchmark functions like the Ackley and the Griewank functions.

Next, we optimize the controller gain of a double inverted pendulum problem. A comparison of results is shown with our proposed method against the P-transform and genetic algorithm.

4.1. **Benchmarking function comparison.** In this section, two common optimization benchmark functions, namely, Ackley [11-13] and Griewank [11,14-16] are used to compare performance against our proposed method and the P-transformation algorithm.

These functions were modified to find the global extremum instead of minimum. We test these two modified functions for 2, 3, 4, 5, 10 and 20 dimensions around the range $[-50, 50]$. Both functions were modified using the following Equation (11)

$$F_{\text{modified}} = F(n) - \text{Original function} \tag{11}$$

where n is the number of variables and F_{modified} is the modified objective function and $F(n) = 10n$.

The Ackley function and the Griewank function are widely used for testing optimization algorithms and are defined by Equations (12) and (13) respectively:

$$f(x) = -a \exp\left(-b\sqrt{\frac{1}{d}\sum_{i=1}^{d} x_i^2}\right) - \exp\left(\frac{1}{d}\sum_{i=1}^{d}\cos(cx_i)\right) + a + \exp(1) \tag{12}$$

$$f(x) = 1 + \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) \tag{13}$$

A comparison of run time results for both modified test functions using the P-transformation and the improved P-transformation is shown in Table 1.

Table 1. Run time comparison of benchmark functions

ACKLEY's Function Dimension	P-transformation Run Time (sec)	Improved P-transformation Run Time (sec)
2	0.290092	0.347802
3	0.536712	0.349026
4	0.382913	0.366289
5	0.682426	0.384799
10	2.091991	0.472864
20	5.8230353	0.334182
GRIEWANK's Function Dimension	P-transformation Run Time (sec)	Improved P-transformation Run Time (sec)
2	0.386757	0.378616
3	0.423401	0.405351
4	0.437406	0.437406
5	0.64111	0.45576
10	0.682507	0.661191
20	1.613999	1.51644

The information above is also depicted in a graphical format as shown in Figures 4 and 5 for the Ackley and Griewanck functions respectively.

As shown on the graph and the table, the run time is improved, particularly for higher dimensions using the improved method as proposed in this paper.

4.2. **Double inverted pendulum application results.** In this section, we demonstrate the application of our proposed method to an inverted double pendulum example on a cart that can move horizontally as shown in Figure 6. The goal is to balance the two sticks in the vertical position by applying a horizontal force to the cart. This is a highly nonlinear problem, which is linearized for this application. This problem is taken from the CADSI/DADS Linear (currently LMS Motion) example handbook [17].

Let x_0 be defined such that both sticks are initially vertical and at rest and the cart starts off with zero displacement and is also at rest while x_c represents the cart position. The linear quadratic regulator control method is used to design the controller and is given by

$$\mathbf{J} = \frac{1}{2}\int_0^{\propto} \left(\delta x(t)^T \mathbf{Q} \delta x(t) + \delta u(t)^T \mathbf{R} \delta u(t)\right) dt \tag{14}$$

In the above equation, x is the system state vector for positions and velocities, u is the system input vector or force and $\mathbf{Q}$ and $\mathbf{R}$ are the weighting matrices. For this example, $\mathbf{Q}$

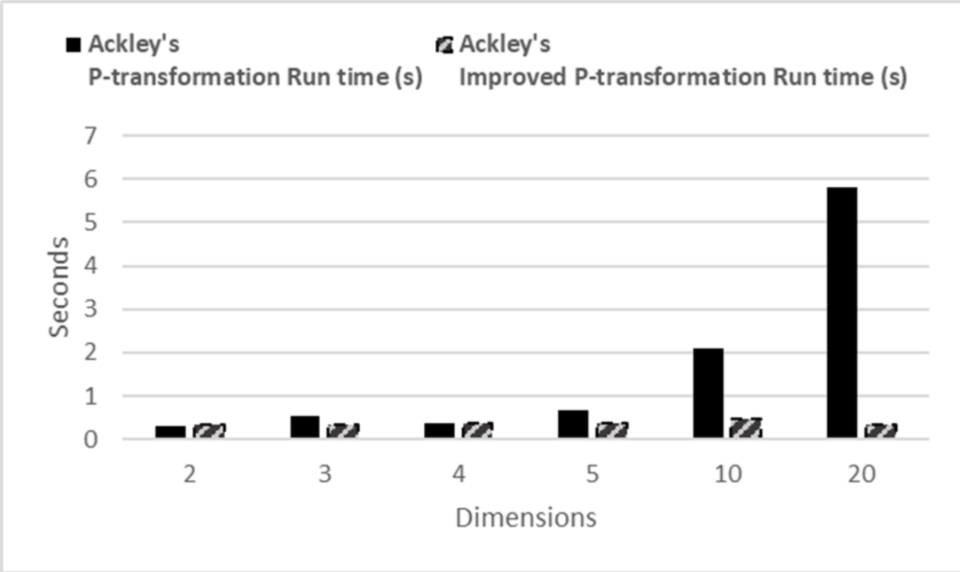

FIGURE 4. Graphical representation of run time results for Ackley function

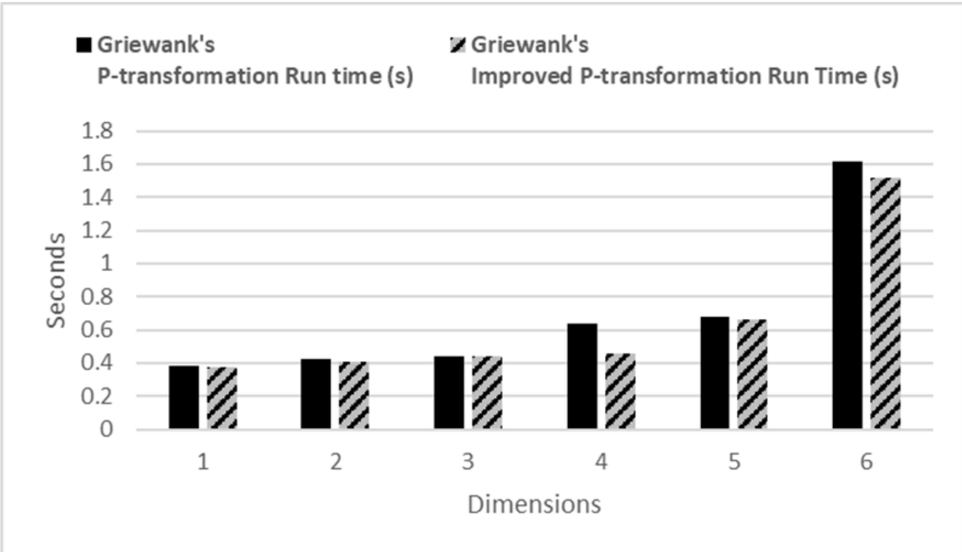

FIGURE 5. Graphical representation of run time results for Griewank function

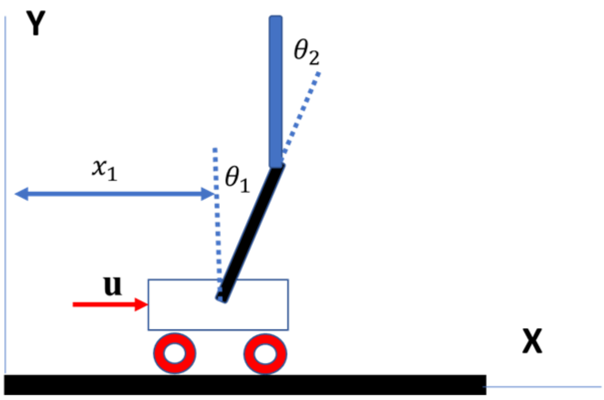

FIGURE 6. Inverted double pendulum example

is a 6×6 identity matrix and $\mathbf{R}$ is a unit scalar vector. Also, δ represents the linearization of the state vectors for the initial condition x_0.

The objective of the problem is to minimize the cost function $\mathbf{J}$ resulting in an input u in terms of the states that keep the sticks vertical. It should be noted that

$$\delta u = -\mathbf{K}\delta x \tag{15}$$

where $\mathbf{K}$ is the gain matrix. By varying the gain $\mathbf{K}$, the cost function $\mathbf{J}$ can be minimized.

Without going into the derivation of the motion equations, the linearized equations for the $\mathbf{A}$ and $\mathbf{B}$ state matrices are taken as indicated in the LMS DADS reference manual and are given by

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 4.999 & -4.292 & 0 & 0 & 0 & 0 \\ -6.66 & 15.51 & 0 & 0 & 0 & 0 \\ 1.867 & -0.2325 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0.5048 \\ -0.6726 \\ 1.179 \end{bmatrix} \tag{16}$$

$$\mathbf{C} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{D} = [0] \tag{17}$$

The state space model is given by

$$\dot{\mathbf{X}} = \mathbf{A}x + \mathbf{B}u \tag{18}$$

$$\mathbf{Y} = \mathbf{C}x + \mathbf{D}u \tag{19}$$

where,

$$\dot{\mathbf{X}} = \begin{bmatrix} \delta\dot{\theta}_1 \\ \delta\dot{\theta}_2 \\ \delta\dot{x}_c \\ \delta\ddot{\theta}_1 \\ \delta\ddot{\theta}_2 \\ \delta\ddot{x}_c \end{bmatrix}, \quad x = \delta x = \begin{bmatrix} \delta\theta_1 \\ \delta\theta_2 \\ \delta x_c \\ \delta\dot{\theta}_1 \\ \delta\dot{\theta}_2 \\ \delta\dot{x}_c \end{bmatrix}, \quad u = \delta u = 0 \tag{20}$$

In the above equation, x_c represents the position of the cart and θ_i represents the angles with the vertical line of each stick.

The optimization problem is posed as follows:
Minimize

$$\sqrt[0.25]{\sum \left(x^2 + (-\mathbf{K}x)^2\right)} \tag{21}$$

Subject to

$$real(sort(eigen(\mathbf{A} - \mathbf{KB}))) + 0.2 \tag{22}$$

where the design variable is the gain $\mathbf{K}$ and its bounds are given by

$$\begin{bmatrix} -100 \\ -200 \\ 0 \\ -100 \\ -100 \\ 5 \end{bmatrix} \leq \mathbf{K} \leq \begin{bmatrix} 10 \\ 10 \\ 5 \\ 5 \\ 1 \\ 10 \end{bmatrix} \tag{23}$$

The MATLAB/SIMULINK state space approach was utilized to solve the problem with the genetic algorithm, the P-transformation, and the improved P-transformation.

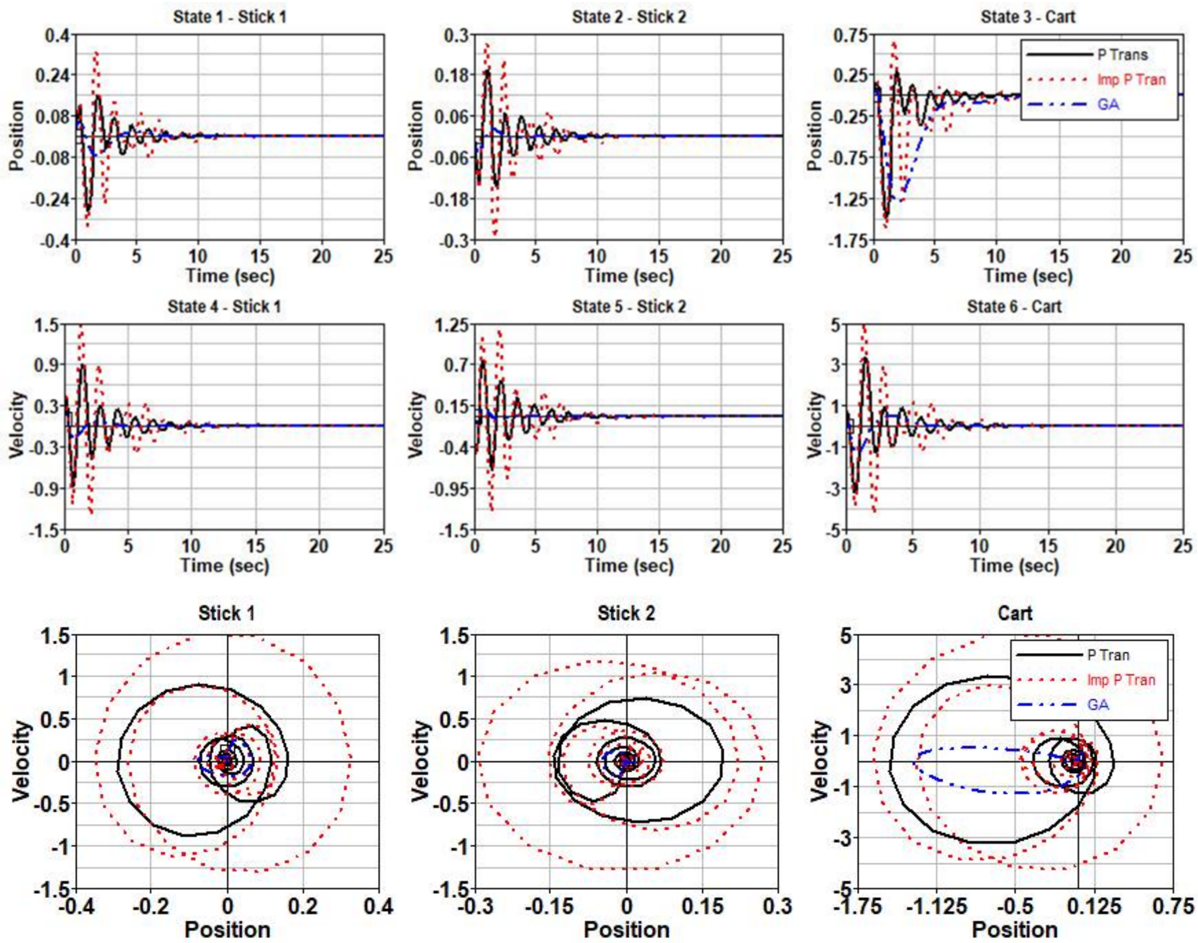

FIGURE 7. Inverted double pendulum states convergence and position vs velocity

A duration of 25 seconds was used for the simulation. The initial velocity for stick 1 was 0.1 with all others as 0. Figure 7 represents the convergence of the states in time and the comparison between position vs velocity.

The optimized values for the gain **K** using the improved P-transformation method are

$$\mathbf{K} = [-78.76 \quad -148.74 \;\; 1.91 \quad -54.05 \quad -29.3 \;\; 8.46]$$

The optimization was run on an 8 core Xeon server with 24Gb RAM. The times utilized in the optimization for the P-transformation, the improved P-transformation, and the genetic algorithm are compared in Table 2.

TABLE 2. Analysis time comparison

Method	Time (sec)
Improved P-transformation	181.02
P-transformation	187.05
Genetic Algorithm	1284.28

We can observe from Table 2 that our proposed algorithm performed the best.

5. **Conclusions.** We have presented the improved P-transformation as an efficient method for global optimization. The examples presented in this paper demonstrate that our proposed algorithm gives better and more efficient results both for the benchmarking functions and a difficult constrained control engineering application for balancing a double pendulum.

The P-transformation is applicable to a wide variety of optimization problems such as: Mixed Integer Nonlinear Problems (MINLP) and cases where the objective functions are absent. There are possibilities that the P-transformation can be utilized hybridly with other global optimization algorithms to expedite computation.

REFERENCES

[1] V. K. Chichinadze, The Ψ-transform for solving linear and nonlinear programming problems, *Automatica*, vol.5, pp.347-355, 1969.

[2] B. Adamczyk, M. A. Zohdy and A. A. Khan, Global optimization of stochastic multivariable functions, *American Control Conference*, San Francisco, 1993.

[3] M. A. Zohdy, A. A. Khan and M. S. Kamel, Global optimization of stochastic multivariable functions, *IEEE Proc. of the 26th Southeastern Symposium on System Theory*, 1994.

[4] V. K. Chichinadze, Solution of nonlinear nonconvex optimization problems by Ψ-transformation method, *Computer and Mathematics with Applications*, vol.21, pp.7-15, 1991.

[5] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*, John Wiley & Sons, 2002.

[6] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, Massachusets, 1989.

[7] C. M. Fonseca and P. J. Fleming, Genetic algorithms for multiple objective optimization: Formulation, discussion and generalization, *Proc. of the 5th International Conference on Genetic Algorithms and Their Application*, Morgan Kaufmann Publishers, San Mateo, CA, USA, 1993.

[8] A. J. Chipperfield, P. J. Fleming and H. Pohlheim, A genetic algorithm toolbox for MATLAB, *Proc. of Int. Conf. Sys. Engineering*, Coventry, UK, pp.200-207, 1994.

[9] M. Livio, *The Golden Ratio: The Story of Phi, the World's Most Astonishing Number*, Broadway Books, New York, 2002.

[10] E. Pap and D. Surla, Lebesgue measure of α-cuts approach for finding the height of the membership function, *Fuzzy Sets and Systems*, vol.111, pp.341-350, 2000.

[11] S. Surjanovic and D. Bingham, *Virtual Library of Simulation Experiments: Test Functions and Datasets*, http://www.sfu.ca/∼ssurjano, Retrieved on October 5, 2017.

[12] X. S. Yang, *Test Problems in Optimization, Engineering Optimization: An Introduction with Metaheuristic Application*, John Wiley & Sons, 2010.

[13] D. H. Ackley, *A Connectionist Machine for Genetic Hill Climbing*, Kluwer Academic Publisher, 1987.

[14] A. O. Griewank, Generalized descent for global optimization, *Journal of Optimization Theory and Applications*, vol.34, pp.11-39, 1981.

[15] M. Locatelli, A note on the Griewank test function, *Journal of Global Optimization*, vol.25, no.2, pp.169-174, 2003.

[16] R. Oldenhuis, *Many Test Functions for Global Optimizers*, Mathworks, 2012.

[17] *DADS Example Manual*, CADSI, Coralville, IA, 1995.

[18] J. Kiefer, Sequential minimax search for a maximum, *Proc. of the American Mathematical Society*, vol.4, no.3, pp.502-506, 1953.