Turkish Journal of Computer and Mathematics Education Vol.12 No.13(2021), 2371-2378

Research Article

Parallel Constrained Predictive Control based on the Improved Particle Swarm Optimization for Nonlinear Fast Dynamic Systems

Supriya P. Diwan[a], Shraddha S. Deshpande[b]

[a] Government College of Engineering, Aurangabad, Maharashtra, India
[b] Walchand College of Engineering, Sangli, Maharashtra, India
[a] Supriya.diwan.etx@geca.ac.in, [b] shraddha.deshpande@walchandsangli.ac.in

Article History: Received: 11 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 4 June 2021

Abstract: As the nonlinear predictive control model (NMPC) has evolved so far, most studies are confined to the slow dynamic nonlinear method, the study difficulty for the general nonlinear systems is mainly derived from optimization algorithm analysis. In fact, most reality control systems are nonlinear and are likely to have limitations. This paper proposed the population selection based improved particle swarm optimization (PS-IPSO) to minimize the computational time of the NMPC algorithm. In the PS-IPSO, the population selection step based on the ranking of population accordance with _tness function evaluation is implemented.

Via simulation results, the improved algorithm's effectiveness is determined by applying it to the highly nonlinear fast dynamic single rotary inverted pendulum (SRIP)system. The solution presented in the paper is computationally feasible for smaller sampling times

Keywords: Nonlinear Model Predictive Control, Particle Swarm Optimization, Fast Dynamic Systems, Rotary Inverted Pendulum, Real-Time Simulation. Particle Swarm Optimization, Population selection, constrained optimization, nonlinear model predictive control

1. Introduction

Model predictive control (MPC),In control systems, model predictive control (MPC) is mostly used as it is efficient and allows the constraints of a system's signals to be taken into account.

Predictive control methods for linear models are well-known. This method cannot be extended directly to nonlinear problems if it is possible to optimize the nonlinear cost function based on an exact, nonlinear framework. Therefore, cost functions, to be reduced at every step, are nonlinear, non-quadratic, and non-convex in general. The estimation of the minimum, the approaches to accurate NMPC use various methods . Due to the nature of the nonlinear system the computation burden is increased which restrict to use of it for slow dynamic processes. To overcome of this problem various methods for solving optimization problems are developed. Sequential Quadratic Programming (SQP) [1], [2] is commonly used for direct methods which mainly depends on the initial point. It is also possible to use successive Linearization (SL) that provides an exact

linear model, but only for a class of nonlinear models, and then to use MPC algorithms [3]. Changing variables in SL, however, can present problems withnonlinear constraints.Genetic Algorithm (GA) optimizers [4] leverage other attempts to solvethe problems of non-convex optimization. However, due to their natural geneticoperations, they face many obstacles, including enormous computationaleffort [5], [6].

The swarm intelligence algorithm is different from many derivative-free optimization techniques in that it is less sensitive to the nature of the objective function, such as consistency and convexity, and iteration does not require good initial solutions. Because of its flexibility, it can be combined with other optimization strategies to create hybrid tools [7]. Because of its simple description and high performance, PSO is a commonly used optimization technique that has been successfully applied to a variety of real-world problems [8] [9]. The beauty of thePSO is its adaptability to changes made in it either by hybridization with otheralgorithms or the modification in itself [14].The key contribution of this paper is to use PS-IPSO based optimization forNMPC algorithm development, resulting in a reduction in overall computationaltime and improved fast dynamic system response. This PS-IPSO based NMPC is applied to the (SRIP) system which is nonlinear fast dynamic system to stabilizethe pendulum position in the inverted direction.

2. NMPC Formulation

In general, Model predictive control, in general, measures control activities repeatedly in order to refine the expected process performance.

2.1 Control relevant Model Selection

The system to be controlled is considered as a nonlinear, state-space model (1) with constraints (3) as follow

x(t+1)=f(x(t),u(t));t≥0, at $t=0$, $x(0)$ (1a)

y(t)=h(x(t),u(t)) (1b)

Depending on the constraints imposed by input and output in the form:

$$u_{min} \leq u(t) \leq u_{max} \qquad (3a)$$

$$y_{min} \leq y(t) \leq y_{max} \qquad (3b)$$

$$\dot{x} = f(x,u) \qquad (3c)$$

Where $x(t) \in \Re^{n_x}$ is the state vector, $u(t) \in \Re^{n_u}$ is the input vector, $y(t) \in \Re^{n_c}$ denotes the controlled output with the t as the current sampling instant. f and h are system functions of the process model. Furthermore, u_{min}, u_{max} and y_{min}, y_{max} are constant vectors. The NMPC's working principle is depicted in Figure 1. (1). A dynamic model of the managed system is used to predict a set of Np potential performance behaviours of the system up to time $t+N_p$ at sample t.i.e., $y(t+N_p|t)$ for $N_p=1,2,\ldots,N_p$. Based on the forecast, N_m optimal future inputs $u(t+N_m|t)$ for $N_m=0,1,\ldots N_m\text{-}1$ are calculated to reach the desired output y_{ref}, as closely as possible as shown in figure 1. The parameters N_p and N_m are the prediction and control horizons respectively.

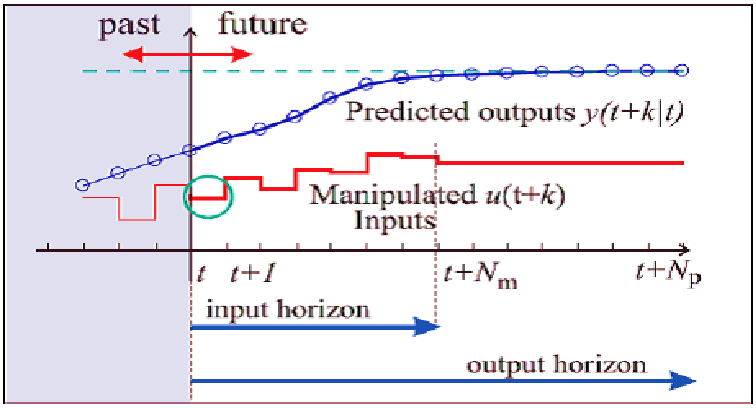

Fig. 1. A graphical representation of NMPC

2.2 Discretization

The continuous model described by the nonlinear state space model is discretized with sample timeT_s. For this, the equations of motion of the pendulum and the rotary arm are defined as equations of difference (4). Provided the notation vector $v(k)$ and sample time T_s, using the forward Euler discretization, the differential system equations are obtained, yielding:

$$x_{e_{k+1}} \approx x_{e_k} + T_s f_e(x_{e_k}, u_{e_k}) = f_d(x_{e_k}, u_{e_k}) \qquad (3)$$

An additional delay state is added to the model to represent the time between when the state variables x_{e_k} are evaluated and when a new control action $u_{e_{k+1}}$ is made available (12).Considering the new state vector $x_k = [x_{e_k}^T x_{u_k}^T]^T$, the input vector$u_k = u_{e_k}$, and $f(x_k, u_k) = \left[f_d(x_{e_k}, x_{u_k})^T \quad u_{e_k}^T\right]^T$, the model takes the form:

$$x_{k+1} = f(x_k, u_k) \qquad (4)$$

2.3 Optimization Problem with Constraints

The optimization problem for NMPC can be de_ned as (5), using the dynamicmodel of form (1).

$$\min_u V(y(t), u(t)) \qquad (5)$$

The deviation between the expected output signal and the target referenceoutput is typically a quadratic function of the minimization criterion for computingthe optimal moves. This cost function V includes the control moves$(u(t+N_m|t))$to minimize the control e_orts. A cost function is in the form of

$$V = \sum_{p=1}^{N_p} ||(y(t+p|t) - y_{ref})||_Q^2 + \sum_{m=0}^{N_m} ||(\Delta u(t+m|t))||_R^2 \qquad (6)$$

Where, Q and R, are weighing matrices. Here, ||.|| is the vector 2-norm, |.| is the absolute value of the vector, and $\Delta u(t+m|t) = u(t+m|t) - u(t+m-1|t)$. Usually , only the first N_m control inputs are calculated, and the following (N_p - N_m) control inputs are assumed to be zero. .Only the first *Nm* control inputs are determined in most situations, andthe remaining *(Np - Nm)* control inputs are considered to be zero. Only thefirst of the Nm control inputs calculated from the minimization of the V isused; the rests are discarded. The output is evaluated at the next samplingmoment, and the process is repeated with the new measured values and bymoving the control and prediction horizons forward. The V is used to determinethe potential optimal control inputs, which can be accomplished using a numberof optimization algorithms.

2.4 Penalty function

The constrained optimization problem (6)(2) is modeled as the NLP problemwith constraints:

$$\min_{u} V(u), \qquad (7a)$$

$$\text{s. t.} \quad h_i(u) \leq 0, i = 1, \ldots., m \qquad (7b)$$

The objective function is V (7a), and the decision vector is u with nu variables.Since an inequality restriction of the form $h_i(u) \geq 0$ may also be interpreted as$-h_i(u) \leq 0$, the formulation of the constraints in (2) is not restrictive. To solvethe constrained optimization problem, the PSO algorithm with penalty functionapproach is implemented in the following equation 8, which is generally definedas:

$$F(u,\sigma) = \begin{cases} V(u) & if\ u\ is\ feasible \\ V(u) + \sigma \sum_{i=1}^{m} [\max\{0, h_i(u)\}]^2 & otherwise \end{cases} \qquad (8)$$

Where for *m* constraints σ is a positive penalty parameter. If *u* is a feasible point, $\max\{0, h_i(u)\} = 0$; else, if u is an unfeasible point, $\max\{0, h_i(u)\} = h_i(u)$. Therefore, the constrained problem is converted into an unconstrained problem:

$$\min_{u} F(u,\sigma)(9)$$

Equation's approximate solution can be found by solving the unconstrainedproblem (7a). The computational efficiency of an NLP problem is determinedby three factors: (1) the size of the problem; (2) the form of problem; and (3) theoptimization algorithm to be used [11]. This paper isn't about the problem styleor the problem size. Because of nonlinear dynamic systems, it is preferable todevelop the optimization algorithm in engineering problems.

In this paper, we proposed a PS-IPSO algorithm to solve the constrainedNMPC optimization problem, which will be discussed in the next session.

3. Proposed PS-PSO

PSO simulates the action of a swarm of birds looking for a food source, forexample. By combining information from each individual, referred to as a particle, with information from the entire colony, the population, or swarm, converges on the best solution [13]. The algorithm starts with a population that is randomly distributed around the design space. From one concept iteration to the next, the position of each particle is changed using the update formula below.

$$x_i^{q+1} = x_i^q + v_i^q \Delta\mathrm{t} \qquad (10)$$

where *i* denotes the ith individual in the swarm, *q* denotes the q^{th} iteration,and *q* denotes the i^{th} individual's velocity vector at the q^{th} iteration. The timeincrement Δt is normally set to unity. At the start, each particle is given arandom velocity vector, which is modified at each iteration using

$$v_i^{q+1} = \omega v_i^q + c_1 r_1 \frac{(p_i - x_i^q)}{\Delta t} + c_2 r_2 \frac{(p^g - x_i^q)}{\Delta t} \qquad (11)$$

where inertiaisω, r_1 and r_2 are random numbers [0,1], and c_1 and c_2 are the parameters which are nothing but the cognitive and social behavior. In addition, The i^{th} particle's best point so far is P_i, while swarm's best point is P_g.The algorithm's search behaviour is regulated by the inertial parameterω, with higher values (around 1.4) indicating a more global search and lower values (around 0.5) indicating a more local search.

Researchers have looked into a number of constraint handling strategies to solve this issue. [11] distinguishes four categories of constraint-handling strategies for evolutionary algorithms: (1) those that maintain viability, (2) those that focus on penalty functions, (3) those that distinguish between feasible and unfeasible solutions, and (4) others.Theconstraint handling in the NMPC algorithm using PS-IPSO is using most popularmethod i.e. penalty cost function. This penalty function is discussed in the

section 2.3.

The time required for the computation in the PSO algorithm is basicallybased on the selection of number of population and the number of iterations.The increased number of population gives the more search space to find the globaloptima which is main objective of the PSO algorithm for the highly nonlinearoptimization problem [12]. In the view of NMPC development for controllingthe real time system, the computation time is the key issue because of the $N_p$and N_m. The total computation of the control input is of based on the N_p*N_m *q. Therefore the modification in the classic PSO algorithm in terms of the

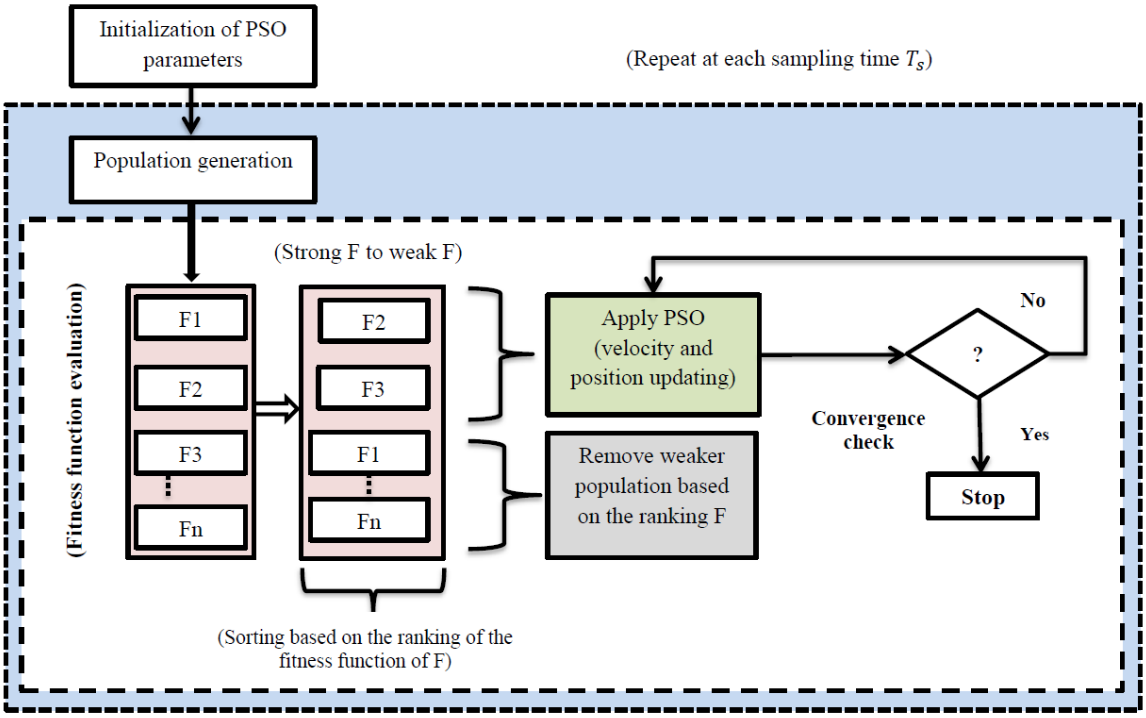

Fig. 2: Strucure of PS-PSO population selection has been proposed.The structure of the PS-IPSO has been given in the (3) and based on the structure the design steps are as follows,

1. The future state variables over the prediction horizon will be generated usingthe nonlinear model (1) and the measured output y from the plant with anestimation of the unmeasured output.
2. Use the predicted state variable to form a nonlinear cost function (6)List the population's solutions according to their fitness (survival of thefittest): The algorithm must be able to decide what makes one solution 'fit'better than another in this phase. The fitness function defines this. The aimof the fitness function is to evaluate viability of the control input in termsof constraints and the cost function minimization (ideally $V = 0$).
3. Cull the weaker solutions:In this step, the algorithm removes the less fitsolutions from the population. Here the population which is near to $V = 0$is selected So the half population will be removed.
4. The PSO algorithm with (10)(11) is applied to remaining population.
5. After completion of the fixed number of iterations, the algorithm computesthe control input U_k
6. Recall that only the first entry u in U_k will be applied to the plant, whereas all other entries are discarded. Thus, it is not necessary to calculate everyentry in U except for the first element u. The calculated first U_{opt} will beapplied to the real-time fast dynamic system.
7. Updated measurements from the plant will be back propagated to the stateupdating (step 1) which will be used to optimized u for appropriate fastoptimization for the next sample (t + 1)

This PS-IPSO based NMPC controller applied to the SRIP which is dicussedin the next section 4.

4. Application of PS-PSO

It is possible to extend this PPSO-based NMPC scheme and its real-time implementationto different systems. In this part, in order to show the superior performanceand effectiveness of the proposed algorithm, we applied the proposedPS-IPSO-based NMPC algorithm to an SRIP system supplied by QUANSER(National

Instruments)SRV02 nonlinear model. The SRIP is chosen as it requiresonline computing efficiency. A simplified control relevant model of SRIP

with 2 degree-of-freedom is used to prove the effective utilization of the proposedalgorithm [15]. The mechanism consists of a rotational arm and a pendulum, with the arm's rotation being regulated by a motor in order to keep the pendulum balanced in an inverted position.There are four states rotational arm position(θ = x_1),pendulumPosition(α = x_2),arm velocity ($\dot{\theta}$= x_3) and the pendulum velocity ($\dot{\alpha}$= x_4). Thereis one input i.e. u is nothing but the voltage applied to the motor which movethe arm in rotational direction to keep pendulum position to required position.A nonlinear model is derived as follows:

$$m_p L_p^2 \, cos(\alpha^2) + J_r)\ddot{\theta} - \left(\frac{1}{2} m_p L_p L_r \, cos(\alpha)\right)\ddot{\alpha} + \left(\frac{1}{2} m_p L_p^2 \, sin(\alpha)\, cos(\alpha)\right)\dot{\theta}\dot{\alpha} + \left(\frac{1}{2} m_p L_p L_r \, sin(\alpha)\right)\dot{\alpha}^2 = \tau - B_r\dot{\theta} \qquad (10a)$$

$$-\frac{1}{2} m_p L_p L_r \, cos(\alpha)\, \ddot{\theta} + \left(J_p + \frac{1}{4} m_p L_p^2\right)\ddot{\alpha} - \frac{1}{4} m_p L_p^2 \, sin(\alpha)\, cos(\alpha)\, \dot{\theta}^2 - \frac{1}{2} m_p L_p g sin(\alpha) = -B_p\dot{\alpha} \qquad (10b)$$

Where the torque applied to the base of the rotary arm (i.e., at the load gear) is generated by a servo motor described by $\tau = \frac{\eta_g K_g \eta_m k_t (V_m - K_g k_m \dot{\theta})}{R_m}$. Based on the above nonlinear dynamic equation (10) the nonlinear state space model is derived where the function f maps the current state and input to the next state$\dot{x}$. This model is used to forecast the state trajectory over a prediction horizon Np and to move the state from a starting condition to a location that is desired with the required control action taken.The samplingtime T_s, which is 20ms, is applied to (12) continuous state space modelnonlinear state space model (4) is derived where the function *f* maps the currentstate and input to the next state [15]. The SRIP begins in a stable equilibriumposition (x_2=*-180 degrees*), and the goal is to invert and stabilize the pendulum

(x_2=*0 degrees*). In this case, the encoder specifies the arm and pendulumpositions. The cost function will be formulated as (6) function (Vm) based on the inverted pendulum controller's requirements for stabilising the inverted pendulum in the upward direction by optimising the control signal with respect to the constraints defined in equation (2) location of input (degree) and input voltage. The design parameters for developing the mathematical model for the SRV02 are described in table 1.

$$-15 \text{ degree} \leq x_2 \leq +15 \text{ degree} \qquad (13a)$$

$$-10\, V_m \leq u \leq +10\, V_m \qquad (13b)$$

Table 1. Parameters of the RIP.

Symbols	Parameters of SRIP	
	Description	**Value**
R_m	Motor armature resistance	2.60 (ohm)
k_t	Motor torque constant	0.00767 (Nm/A)
k_m	Motor back-EMF constant	0.00767 Vs/rad)
K_g	Total gear ratio	70
J_r	Motor armature moment of inertia	0.0010 (kgm^2)
g	Gravitational constant	9.81(m/s^2)
m_p	Pendulum Mass with T-fitting	0.127 (kg)
L_p	Full Length of the pendulum (w/ T-fitting)	0.337 (m)
L_r	Distance from pivot to Centre Of gravity	0.216 (m)
J_p	Pendulum moment of inertia	0.00120 (kgm^2)
B_r	Viscous damping coefficient as seen at the rotary arm axis	0.00240 (Nms/rad)
B_p	Viscous damping coefficient as seen at the pendulum axis	0.00240 (Nms/rad)

The goal of this paper was to use NMPC to control the balance of the SRIP.The values of the various design parameters were initially chosen to provideadequate control efficiency while avoiding unnecessary computational effort.. Thedesign parameters required for the designing of NMPC are mentioned in the table2. In the PS-IPSO for the first iteration fitness function for all the population willbe calculated. Then by arranging the population by ranking of it according tothe fitness function, weaker population (half of the population) will be discarded.Therefore 50 population of swarm out of 100 will be selected for next iteration.Henceforth the optimal control input will be computed as general PSO algorithm.As number of population is reduced to the half with selected population, thecomputational time also reduced.

5. Analysis and simulation results

To demonstrate the performance of the IPSO with population selection approachseveral experiments were performed to _nd the best SRIP optimized with PSIPSO.Generally to control the pendulum at inverted pendulum two controllersare used, one is swing up control and another is balance control [16]. As shownin the simulation results, Fig. 3 the proposed algorithm is able to balance thependulum from its rest position. There is no separate swing up controller is required.

Table 2. Design parameters for the NMPC

Notation	**Value**
N_p	23
N_m	3
T_S	0.02 ms
q	50
i	100
C_1	1.5
C_2	0.5
ω	0.9(max) to 0.4(max)
R	0.1
Q	Diag([1,5])
Tolarance	0.0001

The maximum control input computed by the PS-PSO is approximately2 volts only.

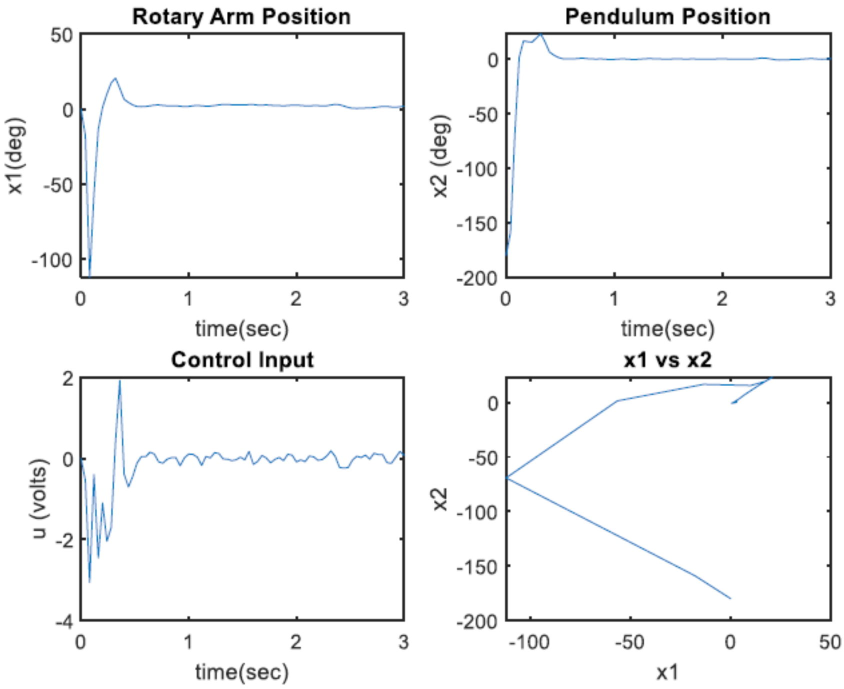

Fig. 3. Simulation result of response of the inverted pendulum from rest position (-180 deg) to the inverted position (stabilization)

For checking the robustness against the disturbance, the step input is appliedas a reference to the rotary arm as shown in the Figure 4.

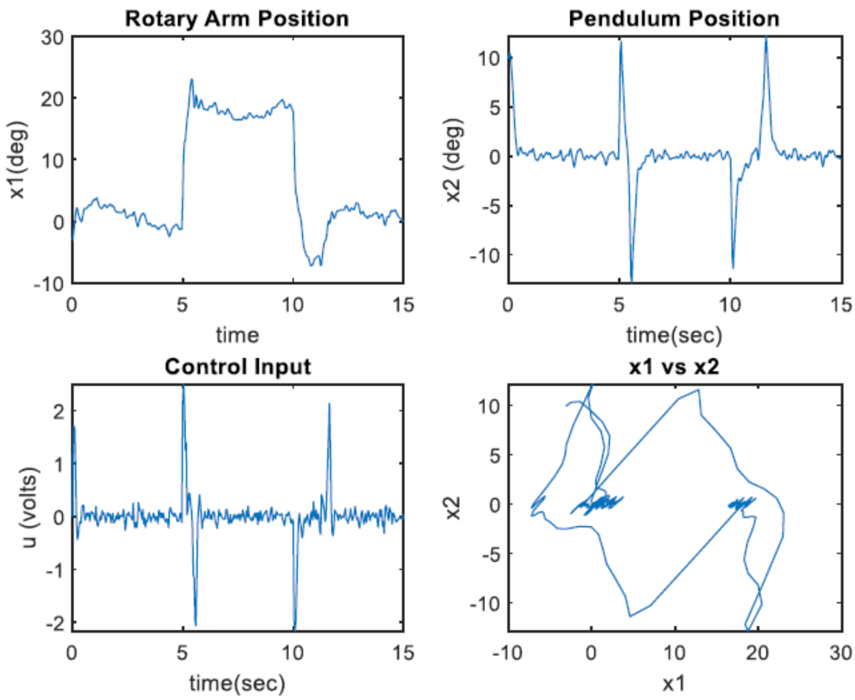

Fig. 4. Simulation result of response of the inverted pendulum while applying step input (as a disturbance) to the rotary arm

The results shows that the inverted pendulum resume to its balance positionquickly. After rigorous iterations the average computational time for thecomputing the optimal input for the SRIP system is 0.01071 second, which isless that the sampling time. The result is compared with the classic PSO algorithmwhich is available in

MATLAB, as GOT(global optimization toolbox).The time required for the GTO is 0.5130 sec which is very large as compared tothe proposed algorithm.

6. Analysis and simulation results

In this paper the proposed PS-IPSO based NMPc algorithm is effectively appliedto the nonlinear fast dynamic system with testing of robustness. All the statesfollowed the prescribed constraints. The computational time is also reduced effciently as it is within sampling time of 20 ms, which shows its efficacy towardsreal-time implementation to control the fast dynamic systems. As the proposedmethod is based on the randomness of the population, results may slightly varywithout much effect of the control action towards system.

The proposed PS-IPSO based NMPC can be used to implement a real-timefast dynamic nonlinear device on a re-configurable (FPGA) embedded platformin the future. Due to the parallel nature of the hardware's processing power, thiscan further decrease the computational time.

References

1. M. Cannon.: E_cient nonlinear model predictive control algorithms.In:In Annual Reviews in Control. volume 28, pages 229-237, 2004. DOI:10. 1016/V.arcontrol.2004 .05.001.
2. M. Diehl.: Optimization algorithms for model predictive control. In : V. Baillieuland T. Samad, editors, Encyclopedia of Systems and Control, pages 1-11. SpringerLondon, London, 2013.
3. V. Zietkiewicz, A. Owczarkowski, and D. Horla. :Performance of feedback linearization
4. based control of bicycle robot in consideration of model inaccuracy. In:Challenges in Automation, Robotics and Measurement Techniques, pages 399-410,Warsaw, 2016. DOI: 10.1007/978-3-319-29357-8-36.
5. X. Blasco, M. Martinez, V. Senent, and V. Sanchis.:Generalized predictive controlusing genetic algorithms (GAGPC): an application to control of a non-linearprocess with model uncertainty. In: Methodology and Tools in Knowledge-BasedSystems, ser. Lecture Notes in Computer Science. Springer, 1998, pp. 428-437.
6. T. Kawabe and T. Tagami.:A real coded genetic algorithm for matrix inequalitydesign approach of robust PID controller with two degrees of freedom. In: Proceedingsof the 1997 IEEE International Symposium on Intelligent Control, Vul 1997,pp. 119-124.
7. R. Krohling, H. Vaschek, and V. Rey.:Designing PI/PID controllers for a motioncontrol system based on genetic algorithms.In: Proceedings of the 1997 IEEE InternationalSymposium on Intelligent Control, Vul 1997, pp. 125-130.
8. M. R. AlRashidi and M. E. El-Hawary.:A survey of particle swarm optimizationapplications in electric power systems.:Evolutionary Computation, IEEE Transactionson, vol. 13, no. 4, pp. 913 -918, aug. 2009.
9. R. Eberhart and Y. Shi.:Particle swarm optimization: developments, applicationsand resources.In: Proceedings of the 2001 Congress on Evolutionary Computation,vol. 1, 2001, pp. 81-86.
10. X. Hu, Y. Shi, and R. Eberhart.:Recent advances in particle swarm.In: Congresson Evolutionary Computation, CEC2004, vol. 1, Vune 2004, pp. 90-97.
11. C. Coello, G. Pulido, and M. Lechuga.:Handling multiple objectives with particleswarm optimization.In:IEEE Transactions on Evolutionary Computation, vol. 8,no. 3, pp. 256-279, Vune 2004.
12. G. Coath and S. Halgamuge.:A comparison of constraint-handling methods for the
13. application of particle swarm optimization to constrained nonlinear optimizationproblems. In:Evolutionary Computation. CEC '03. The 2003 Congress on, vol. 4,dec. 2003, pp. 2419 - 2425.
14. I. El-Gallad, M. E. El-Hawary, and A. A. Sallam.In: Swarming of intelligent particles
15. for solving the nonlinear constrained optimization problem. In: InternationalVournal of Engineering Intelligent Systems, vol. 9, no. 3, pp. 155-163, 2001.
16. V. Kennedy and R. Eberhart. : Particle swarm optimization. In: Neural Networks,1995. Proceedings., IEEE International Conference on, volume 4, pages 1942-1948,1995.
17. Zhang Y., S. Wang, G. Vi.: A Comprehensive Survey on Particle Swarm Optimization Algorithm and Its Applications.In: Mathematical Problems in Engineering, vol. 2015, Article ID 931256, 38 pages, 2015. https://doi.org/10.1155/2015/931256
18. Diwan, S.P., Deshpande, S.S.: Nonlinear Model Predictive Controller for the Real-Time control of Fast Dynamic System, in: Proceedings of the 4th International Conference on Communication and Electronics Systems, ICCES 2019. https://doi.org/10.1109/ICCES45898.2019.9002380
19. Astrom, K.V., Furuta, K.: Swinging up a pendulum by energy control. In: Automatica 36,2000, https://doi.org/10.1016/S0005-1098(99)00140-5