



Kinematic-Model-Free Predictive Control for Robotic Manipulator Target Reaching With Obstacle Avoidance

Ahmad AlAttar^{1,2*}, Digby Chappell¹ and Petar Kormushev¹

¹Robot Intelligence Lab, Dyson School of Design Engineering, Imperial College London, London, United Kingdom, ²Robotics Lab, Dubai Future Labs, Dubai, United Arab Emirates

OPEN ACCESS

Edited by:

Wei-Yun Yau,
Institute for Infocomm Research
(A*STAR), Singapore

Reviewed by:

Tairen Sun,
Jiangsu University, China
Thang Tien Nguyen,
Texas A&M University Corpus Christi,
United States

*Correspondence:

Ahmad AlAttar
a.alattar19@imperial.ac.uk

Specialty section:

This article was submitted to
Robotic Control Systems,
a section of the journal
Frontiers in Robotics and AI

Received: 04 November 2021

Accepted: 07 January 2022

Published: 02 February 2022

Citation:

AlAttar A, Chappell D and
Kormushev P (2022) Kinematic-
Model-Free Predictive Control for
Robotic Manipulator Target Reaching
With Obstacle Avoidance.
Front. Robot. AI 9:809114.
doi: 10.3389/frobt.2022.809114

Model predictive control is a widely used optimal control method for robot path planning and obstacle avoidance. This control method, however, requires a system model to optimize control over a finite time horizon and possible trajectories. Certain types of robots, such as soft robots, continuum robots, and transforming robots, can be challenging to model, especially in unstructured or unknown environments. Kinematic-model-free control can overcome these challenges by learning local linear models online. This paper presents a novel perception-based robot motion controller, the kinematic-model-free predictive controller, that is capable of controlling robot manipulators without any prior knowledge of the robot's kinematic structure and dynamic parameters and is able to perform end-effector obstacle avoidance. Simulations and physical experiments were conducted to demonstrate the ability and adaptability of the controller to perform simultaneous target reaching and obstacle avoidance.

Keywords: kinematic-model-free, predictive control, target reaching, obstacle avoidance, adaptive control, model-free control

1 INTRODUCTION

Researchers over the past few decades studied optimal control, especially for applications where systems dynamics and constraints require proper handling (Zanon et al., 2017). Model Predictive Control (MPC) is an advance control method widely used in academia and industry (Forbes et al., 2015). MPCs are widely used in robotics due to their ability to handle constraints and run in real-time (Saputra et al., 2021). In general, MPCs require a system model prior to control in order to optimize the controller over a finite horizon (Stenman, 1999) which can be challenging to obtain or maintain for certain types of robots (e.g., soft robots, continuum robots, transforming robots, etc.) (AlAttar et al., 2021). The kinematic-model-free (KMF) controller is capable of controlling robot manipulators without any previous knowledge of the robot's kinematic structure or dynamic properties. The controller works by building local linear models that are used to perform reaching tasks (AlAttar and Kormushev, 2020). However, the KMF controller up to date have not been applied to high-level tasks such as trajectory planning and obstacle avoidance as they are concerned with low-level control. Along with the merits of MPCs, integrated together, they can perform high-level tasks. In this paper, a novel kinematic-model-free predictive controller (KMF-PC), that is capable of performing end-effector reaching and obstacle avoidance for robot manipulators, is presented. The controller was tested in simulation as well as on a physical 2-degrees-of-freedom (2-DOF) manipulator shown in **Figure 1**.

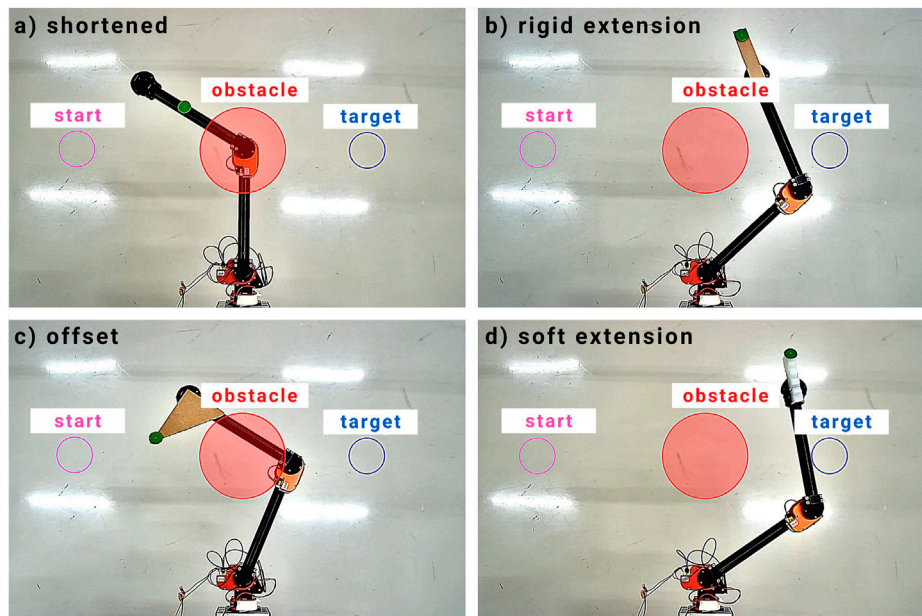


FIGURE 1 | Experimental setup to test the adaptability of the proposed KMF-PC controller. The magenta circle represents the initial point prior to robot motion, the red circle represents a virtual obstacle, and the blue circle represents the target position. The green circle on the robot is the tracked point. Midway reaching, the end effector is modified to demonstrate the adaptability of the controller. Four tests were conducted: **(A)** The end effector was shortened by moving the tracked point down the second link; **(B)** The end effector was lengthen by adding a rigid extension; **(C)** The end effector was offset at an angle; **(D)** The end effector was extended using a soft link.

1.1 Contributions

The contributions of the paper are: Presenting a novel model-free controller, KMF-PC, capable of controlling a robot arm without any prior knowledge of the robot's kinematics or dynamics and achieving end-effector obstacle avoidance using a kinematic-model-free approach. Furthermore, the adaptability of the controller is tested by modifying the end effector during execution.

1.2 Paper Structure

The paper is organized as follows: **Section 2** introduces work in the literature that is relevant to the research presented, **Section 3** presents the formulation of the KMF-PC controller, **Section 4** presents simulation and experimental results, and finally **Section 5** concludes the paper.

2 RELATED WORK

MPC is a control method that uses a system model to predict future behavior which can be sensitive to prediction errors, especially in the case of nonlinear plants. To mitigate such sensitivities, adaptive model predictive control, where the plant model and nominal conditions are continuously adapted, is used (Franco and Santos, 2019). MPCs gained popularity due to its ability to handle constraints and its capacity to run in real-time, which allowed it to be widely used in robotics (Saputra et al., 2021). MPCs have been used for path planning and collision avoidance of road vehicles (Ji et al., 2016) and

multiple micro aerial vehicles (Kamel et al., 2017). Krämer et al. (2020) controlled a manipulator to perform reaching tasks while avoiding a dynamic obstacle. Nikou et al. (2017) used a nonlinear-MPC to control N robots for cooperative manipulation of objects whilst avoiding obstacles. Li and Xiong (2019) performed obstacle avoidance of mobile manipulators using an MPC.

Li et al. (2021) used MPC for trajectory tracking and obstacle avoidance of mobile robots. Oleinikov et al. (2021) used MPC for controlling a robot manipulator that avoids fixed and human obstacles. Ashe and Krishna (2020) used MPC for dynamic target tracking and obstacle avoidance for a person-following robot. Pankert and Hutter (2020) used MPC for continuous mobile manipulation and collision avoidance. Other researchers used MPC for controlling an autonomous vehicle and performing obstacle avoidance (Quirynen et al., 2020; Song and Huh, 2021). Wang et al. (2021) used a MPC for controlling a dog robot for helping vision-impaired people to navigate around obstacles.

In general, MPCs require a system model for predicting the state of the system over a finite horizon Dao et al. (2021). This imposes that the system to be controlled and its dynamic model must be known prior to control. Furthermore, nonlinear models can burden the controller with heavy computational requirements (Stenman, 1999). This can be challenging in robotics as some robots, such as soft and continuum robots, are not straightforward to model accurately, while others might change, adapt, morph, transform, or evolve, such as robots that change their kinematic structure or dynamic properties during

movement or robots that get accidentally damaged or bent (AlAttar et al., 2021).

On the other hand, KMF control methods can overcome these challenges as these control methods do not require any prior knowledge of the robot’s kinematic or dynamic models. Kinematic-model-free controllers work by gathering information through short exploratory actuations, building a local linear model that approximates the robot’s local kinematic and dynamic behaviour, and estimating an actuation signal that moves the robot towards a given reference target. In our previous work, we’ve demonstrated KMF position control of a robot manipulator capable of performing target-reaching tasks (Kormushev et al., 2015a, Kormushev et al., 2015b). The controller was then experimented on a pneumatically-actuated soft continuum robot (Frazelle et al., 2021) for space applications. We, then, presented a pose (position and orientation) controller that controls a robot’s end-effector pose using locally weighed dual quaternions (AlAttar and Kormushev, 2020). Finally, we scaled up the degrees-of-freedom to be controlled by tracking and controlling multiple points along the robot’s kinematic chain (AlAttar et al., 2021).

Although model-free predictive control methods were previously studied (Stenman, 1999; Yamamoto, 2014; Li and Yamamoto, 2016), online kinematic-model-free predictive control for robot manipulators is an unexplored area of research up to our knowledge.

3 PROBLEM FORMULATION

This section formulates the KMF-PC controller. The nonlinear model of a rigid-body robot manipulator can be obtained using the following general dynamic equation:

$$M(q(t)\ddot{q} + c(q, \dot{q}) + g(q(t)) + \tau_f(t) = \tau(t) \tag{1}$$

where q are the joint angles, \dot{q} are the joint velocities, \ddot{q} are the joint accelerations, M is the inertia matrix, c is the Coriolis vector, g is the gravity vector, τ_f is the friction torque, and τ is the applied torque.

3.1 Kinematic-Model-Free Control

For a 2-DOF robot manipulator, let \hat{p} be the desired actuation primitive whose parameters $\tau_p(\hat{p})$ are to be estimated to actuate the end effector towards the target position:

$$b_1 = \begin{bmatrix} \tau_p^1(\hat{p}) \\ \tau_p^2(\hat{p}) \end{bmatrix} \tag{2}$$

Following the kinematic-model-free approach, b_1 can be approximated as a linear combination of actuation primitives as follows:

$$A_1 w = b_1 \tag{3}$$

where $w = [w_1, w_2, \dots, w_k]^T$ is the vector of unknown weights and A_1 , the actuation matrix, is defined as follows:

$$A_1 = \begin{bmatrix} \tau_p^1(p_1) & \tau_p^1(p_2) & \dots & \tau_p^1(p_k) \\ \tau_p^2(p_1) & \tau_p^2(p_2) & \dots & \tau_p^2(p_k) \end{bmatrix} \tag{4}$$

During exploration, the controller applies pseudo-random actuation primitives with the following curve:

$$\tau(t) = \begin{cases} \tau_p & t \in \left[t_0, t_0 + \frac{d_p}{2} \right) \\ 0 & otherwise \end{cases} \tag{5}$$

The resulting end-effector displacements are recorded into the observation matrix:

$$A_2 = \begin{bmatrix} \Delta x(p_1) & \Delta x(p_2) & \dots & \Delta x(p_k) \\ \Delta y(p_1) & \Delta y(p_2) & \dots & \Delta y(p_k) \\ \Delta z(p_1) & \Delta z(p_2) & \dots & \Delta z(p_k) \end{bmatrix} \tag{6}$$

where $\Delta i(p_j)$ is the displacement effect on the end effector along the i -axis as a result of exerting actuation p_j . In the planar case, $\Delta z(p_j) = 0$. Let the desired effect be expressed as follows:

$$b_2 = \begin{bmatrix} \Delta x(\hat{p}) \\ \Delta y(\hat{p}) \\ \Delta z(\hat{p}) \end{bmatrix} \tag{7}$$

The desired effect, b_2 , can be expressed as a linear combination of the kNN observed effects, A_2 , as follows:

$$A_2 w = b_2 \tag{8}$$

Note that many solutions might exist for w since A_2 is not full rank. To resolve this issue, least squares regression is used to find the smallest squared vector b_2 that is a solution to the equation. This calculated value can then be used in Eq. 3 to find the desired actuation primitive that will actuate the end effector towards the desired target position. After every actuation, the actual and desired end-effector positions are compared. In the case of significant discrepancy, another exploratory phase is triggered.

3.2 Model Predictive Control

The system model of linear MPC in state space form is as follows:

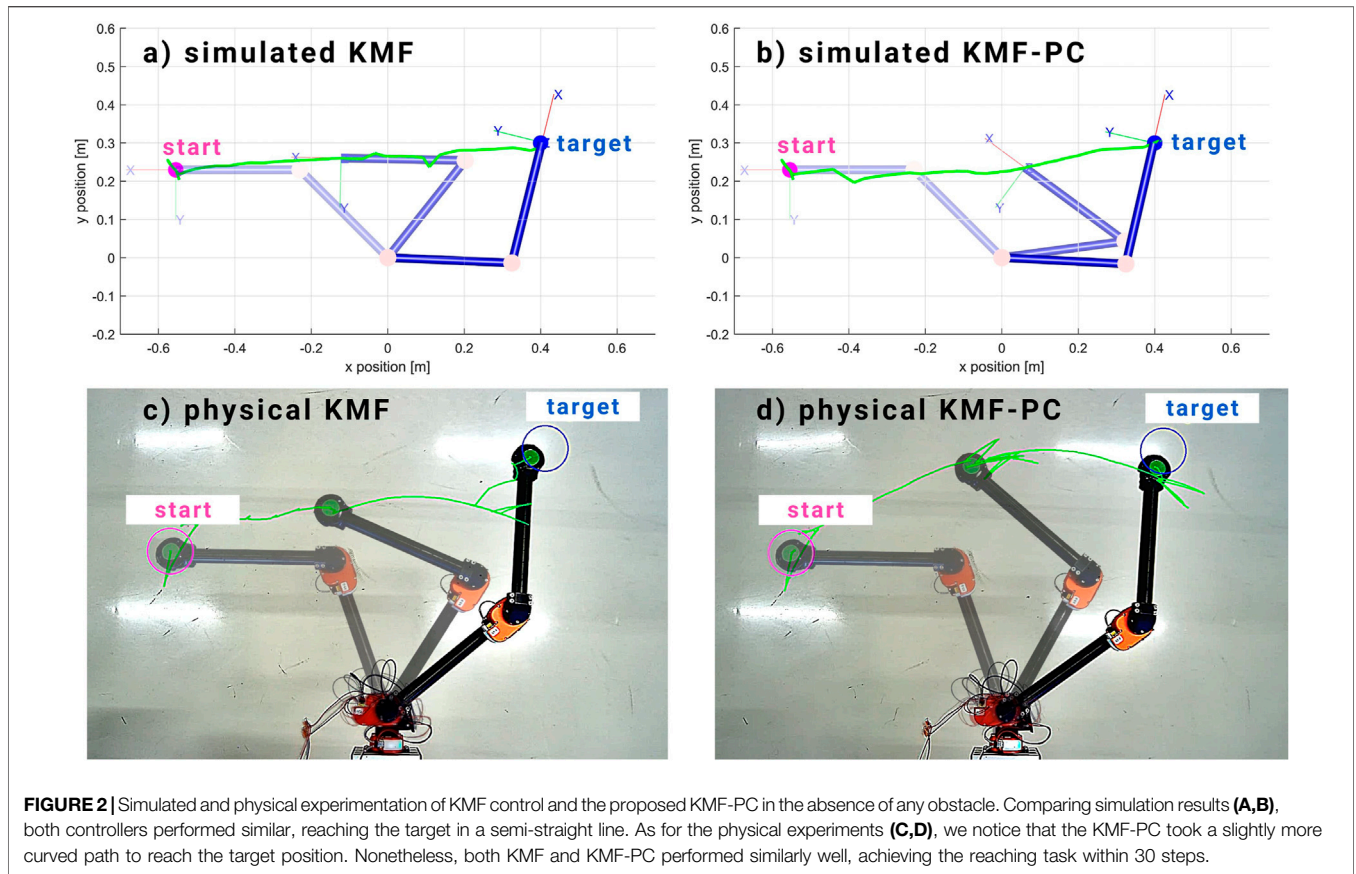
$$x(k+1) = Ax(k) + Bu(k) \tag{9}$$

$$y(k) = Cx(k) + Du(k) \tag{10}$$

where matrices A , B , C , and D can vary with time and:

- k: current control time index
- x: system state variables
- u: manipulable inputs
- y: system outputs

In general, a system model is used to predict the future states of a dynamic system by the MPC to generate a control signal that minimizes some cost function over a predefined prediction horizon (Afram and Janabi-Sharifi, 2014). The cost function optimizes the control variables to achieve some desirable reference which results in a sequence of N optimal actuations, of which the MPC only applies the first (Khosravi et al., 2019).



This process is then repeated after shifting the horizon forward. If the system model is linear, analytical solutions to the MPC can be used for extremely fast computation. In the case of rigid-body robots, the dynamic model shown in (Eq. 1) is integrated in time, and becomes extremely non-linear as more joints are added, making MPC challenging. The problem is often simplified by linearizing the system dynamics, however this requires an accurate model of the system; something that is not possible for unknown robots.

3.3 Kinematic-Model-Free Predictive Control

To make use of linear MPCs, a state space formulation for KMF control is required. The actuation equation in KMF control can be premultiplied by the pseudoinverse of A_1 , the actuation matrix:

$$w = A_1^\dagger b_1. \tag{11}$$

This can then be substituted into the observation equation to yield the following:

$$b_2 = A_2 A_1^\dagger b_1 \tag{12}$$

Since $\Delta x(k+1) = b_2$ and $u(k) = b_1$, we can easily notice that:

$$\Delta x(k+1) = Bu(k) \tag{13}$$

where

$$B = A_2 A_1^\dagger \tag{14}$$

The cost function is defined as follows:

$$l(x, u) = \|x_{des} - x_u\| + \|u\| \tag{15}$$

where x_{des} is the desired state of the system. Formally, the optimization problem is expressed as follows:

$$\min_u J_N(x_0, u) = \sum_{k=0}^{N-1} l(x(k), u(k)) \tag{16}$$

$$s.t. \quad \Delta x(k+1) = Bu(k) \tag{17}$$

$$x(0) = x_0 \tag{18}$$

$$\|u\| \leq u_{max} \tag{19}$$

$$(x_1(k) - a)^2 + (x_2(k) - b)^2 \geq r^2 \tag{20}$$

where x_0 is the initial state of the system, u_{max} is the maximum torque to be applied, $x_i(k)$ is the i th system state variable, and $\{a, b, r\}$ are the circular constraint parameters of the obstacle. Since the local linear model built is volatile and is only used for the next actuation, a linear MPC suffices. As with all KMF controllers, stability and convergence are not guaranteed due to their exploratory nature.

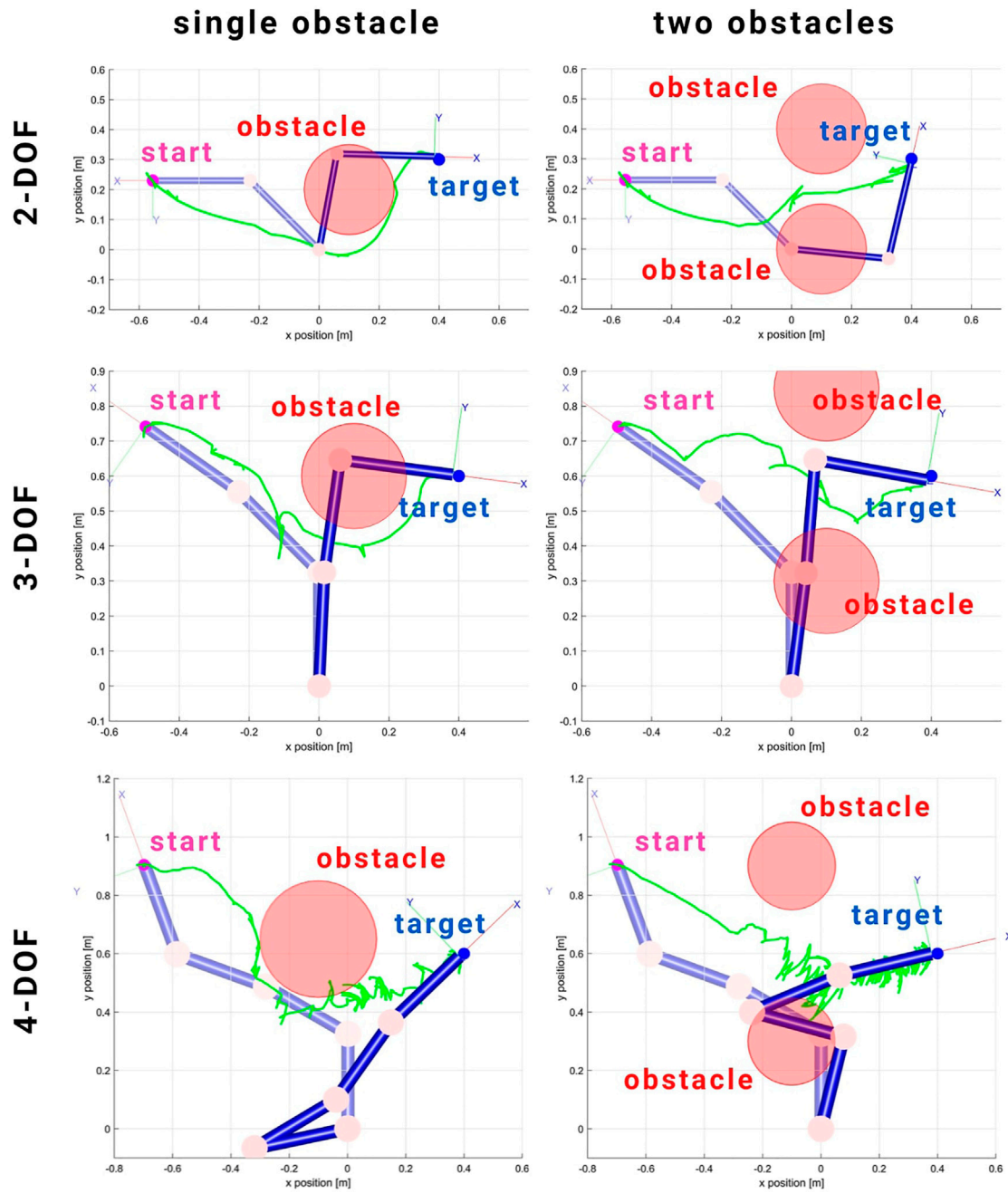


FIGURE 3 | Simulation results of simultaneous reaching and obstacle avoidance using KMF-PC on simulated 2-DOF, 3-DOF, and 4-DOF manipulator robots. It is noted in the 4-DOF case, where the complexity is relatively higher, more exploratory motion is triggered which caused the end effector to penetrate the virtual obstacle.

4 RESULTS

In this section, simulation and physical experimentation results are presented and discussed. In the first experiment, a simple reaching task on a simulated 2-DOF robot and a physical 2-DOF robot was performed using classical KMF controller and the proposed KMF-PC. In the second experiment, reaching tasks on a 2-DOF, 3-DOF, and 4-DOF simulated robot as well as the 2-DOF physical robot was performed with one and two

circular obstacles. In the third experiment, the adaptability of the KMF-PC is demonstrated by modifying the 2-DOF robot’s kinematic structure midway. All the robots are planar, restricting the motion to the horizontal plane, to neglect the effects of gravity.

The simulation robots are simulated on MATLAB using Peter Corke’s Robotics Toolbox (Corke, 2017). The KMF-PC makes use of CasADi (Andersson et al., 2019). End-effector position was easily obtained using toolbox functions. On the

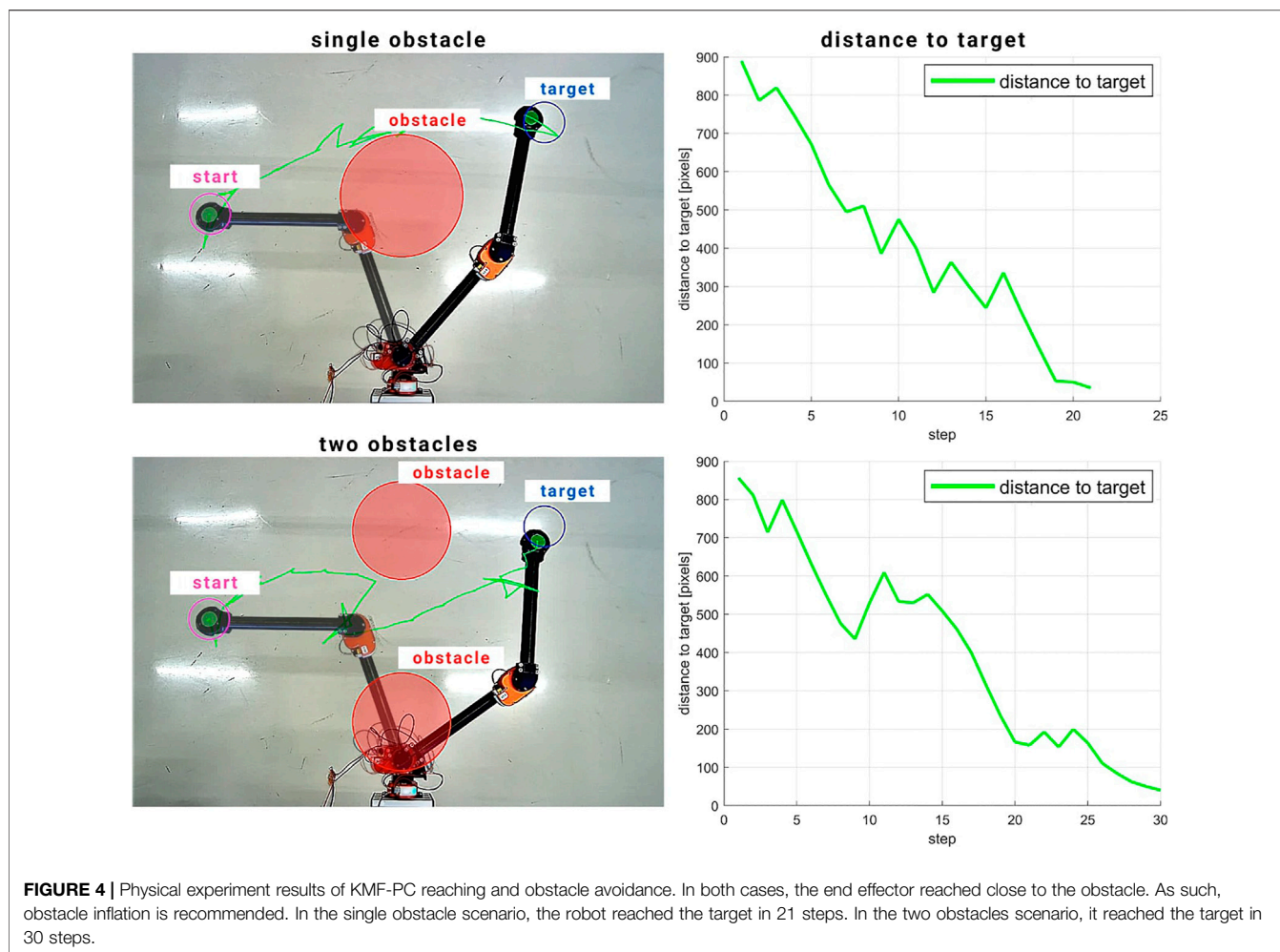


FIGURE 4 | Physical experiment results of KMF-PC reaching and obstacle avoidance. In both cases, the end effector reached close to the obstacle. As such, obstacle inflation is recommended. In the single obstacle scenario, the robot reached the target in 21 steps. In the two obstacles scenario, it reached the target in 30 steps.

other hand, the physical robot is comprised of aluminium links and Hebi X-Series actuators, (X5-9) capable of delivering 9 Nm of continuous torque, which are controlled using torque signals. The end effector of the robot is tracked using a calibrated webcam and a green marker. Direct pixel coordinates is used for determining the end-effector position for physical experiments. Virtual obstacles are used to emulate a circular obstacle that is to be avoided.

4.1 Experiment 1: Reaching Using Classical KMF and KMF-PC

In this experiment, classical KMF and the proposed KMF-PC are used to perform a simple reaching task on a simulated and a physical 2-DOF robot in the absence of any obstacle. **Figure 2** shows both the simulated and physical experimentation results of running both controllers.

It is evident that, in the absence of any obstacle, both controllers performed similarly and were able to reach the target position in less than 30 steps.

4.2 Experiment 2: Obstacle Avoidance Using KMF-PC

In the second experiment, obstacle avoidance is performed on a 2-DOF, 3-DOF, and 4-DOF simulated robot and the 2-DOF physical robot. The results of simulations are summarized in **Figure 3**. The KMF-PC is capable of avoiding obstacles along the reaching path. Despite the increase in complexity when an extra DOF is added, the controller successfully performs the reaching task whilst avoiding the obstacles (2-DOF and 3-DOF). It is also noted, due to the exploratory nature of the KMF approach, should the controller trigger an exploratory phase near any obstacle, this might cause collision with the obstacle (evident in the 4-DOF case). As such, it is recommended to inflate the obstacles to avoid any collision due to exploratory behaviour.

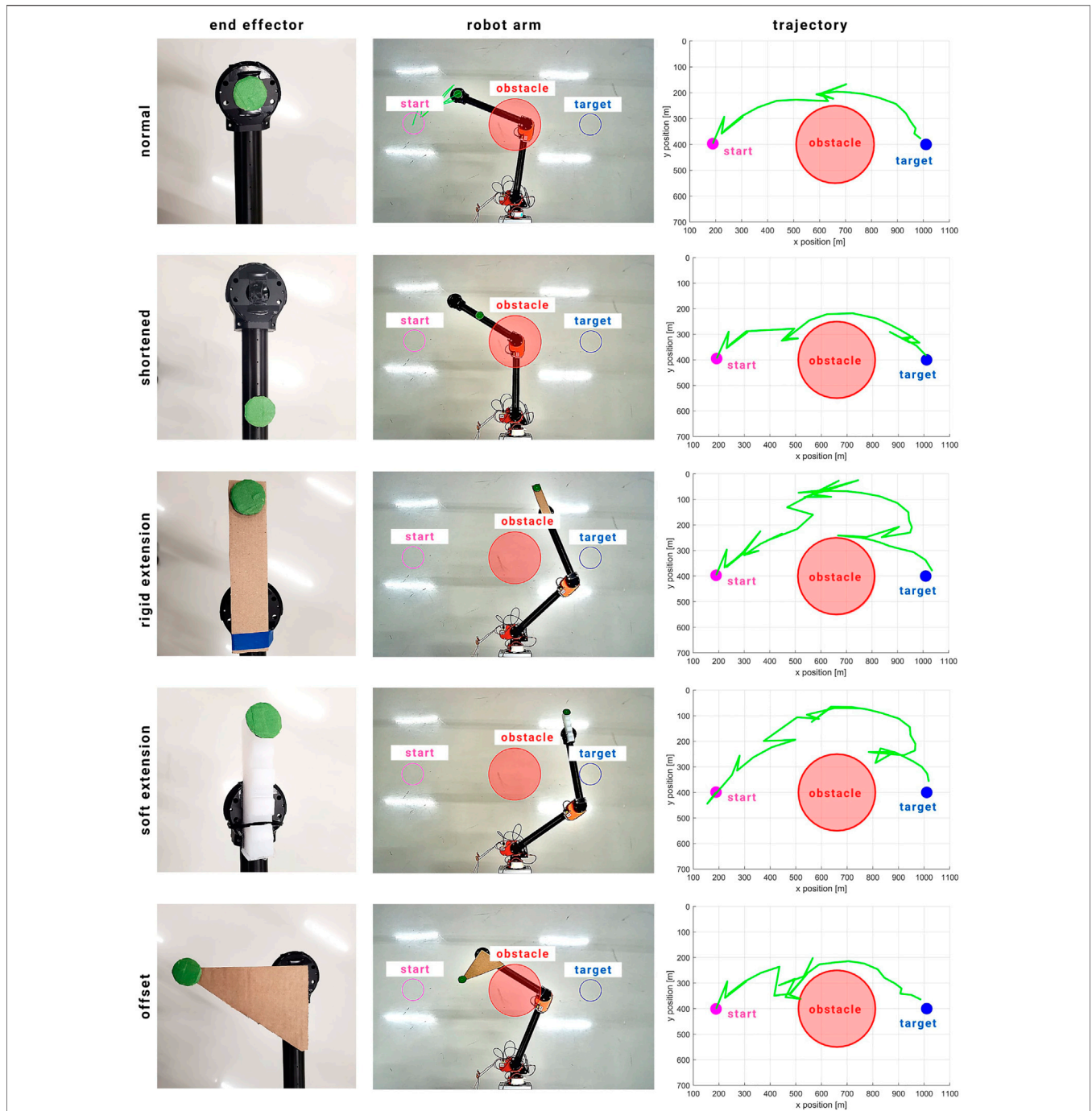


FIGURE 5 | Physical experiment results of reaching and obstacle avoidance whilst modifying the end effector during execution. When the end effector is altered, the controller triggers an exploration phase and resumes reaching and obstacle avoidance right after. The modifications include shortening the end effector, rigid and soft extensions, and offsetting the end effector.

As for the physical experiments, the results are shown in **Figure 4**. The KMF-PC controller successfully performed the reaching task whilst avoiding the obstacles. In both cases, single

and two obstacles, the KMF-PC controller brought the instances where the robot came close to the obstacle(s). As such, inflating the obstacles is recommended.

When one obstacle was presented, the controller took 21 steps to reach the target end-effector position. When two obstacles were in the way, the controller took 30 steps to reach the target.

4.3 Experiment 3: Adaptability of KMF-PC

In the third experiment, the adaptability of the proposed controller is demonstrated through a series of tests in which the kinematic structure of the physical robot is modified midway the reaching task. The results are summarized in **Figure 5**. First, the robot's second link is shortened by moving the marker down the link. Second, the link is extended by adding a rigid cardboard to the last link. Third, the end effector is extended using a soft link. Fourth, the end effector is offset.

The end effector is altered by pausing the execution temporarily and manually modifying the end effector. After the modification, the execution is resumed. In all cases, when the execution is resumed, the controller goes into an exploratory phase which is triggered due to the discrepancy measured between the predicted and actual state of the end effector. The controller, then after, resumes moving toward the target position. In all cases, the robot managed to reach the target within 50 steps.

5 CONCLUSION

A novel KMF controller is proposed that is capable of obstacle avoidance, using an MPC approach, without any prior knowledge of the robot's model. KMF controllers make no assumption about the robot's structure nor the environment. The formulation of the proposed KMF-PC was derived in this paper, showing how obstacle avoidance can be achieved without prior knowledge of the robot's dynamics or kinematics. Using this approach to allow MPC to be applied to unknown robots enables a greater level of control to be achieved, with minimal computational complexity. Both simulated and physical experiments were conducted. Running the KMF-PC controller in the absence of any obstacle behaved similar to the classical KMF controller. Single and double obstacle avoidance was tested in simulation using 2-DOF, 3-DOF, and 4-DOF robots and on a physical 2-DOF robot. It is found that due to the exploratory nature of KMF control, any exploratory actuations trigger near any obstacle might cause the robot's end effector to penetrate or collide with the obstacle. As such, obstacle inflation is recommended. Furthermore, the adaptability of the KMF-PC was demonstrated

REFERENCES

- Afram, A., and Janabi-Sharifi, F. (2014). Theory and Applications of HVAC Control Systems - A Review of Model Predictive Control (MPC). *Building Environ.* 72, 343–355. doi:10.1016/j.buildenv.2013.11.016
- AlAttar, A., Cursi, F., and Kormushev, P. (2021). Kinematic-model-free Redundancy Resolution Using Multi-point Tracking and Control for Robot Manipulation. *Appl. Sci.* 11, 4746. doi:10.3390/app11114746
- AlAttar, A., and Kormushev, P. (2020). Kinematic-model-free Orientation Control for Robot Manipulation Using Locally Weighted Dual Quaternions. *Robotics* 9, 76. doi:10.3390/robotics9040076

through a series of tests where the end effector was modified halfway during execution. In all experiments, the robot managed to perform simultaneous reaching and obstacle avoidance, reaching the target position in less than 50 steps. At the current state, KMF should be applied to robot control problems where flexibility and adaptability is favoured over accuracy and speed. Future work includes obstacle-aware exploratory actuations that take into account the obstacles to avoid collisions and continuous control to allow for gravity compensation and, thus, non-planar control.

DATA AVAILABILITY STATEMENT

The datasets presented in this article are not readily available because the data generated was from simulation runs and physical experimentations on the robot. The data collected are summarized in figures in the article. Requests to access the datasets should be directed to a.alattar19@imperial.ac.uk.

AUTHOR CONTRIBUTIONS

AA worked on the kinematic-model-free part, ran the simulations, conducted the physical experiments, and wrote the paper. DC worked on integrating KMF with MPC theory and reviewed the paper. PK supervised the research and reviewed the paper.

FUNDING

AA was supported by the Ministry of Education of the United Arab Emirates through their scholarship program and this research was supported by Dubai Future Labs through their facility.

ACKNOWLEDGMENTS

I wish to thank Dr. Tarek Taha for his support with networking and communication setup. I would also like to thank the technical team (Majed, Alex, Julian, and Layth) at Dubai Future Labs for their support throughout the setup and experimentation.

- Andersson, J. A. E., Gillis, J., Horn, G., Rawlings, J. B., and Diehl, M. (2019). CasADi: a Software Framework for Nonlinear Optimization and Optimal Control. *Math. Prog. Comp.* 11, 1–36. doi:10.1007/s12532-018-0139-4
- Ashe, A., and Krishna, K. M. (2020). "Dynamic Target Tracking & Collision Avoidance Behaviour of Person Following Robot Using Model Predictive Control," in 2020 24th International Conference on System Theory, Control and Computing (ICSTCC), 8–10 Oct. 2020 (IEEE), 660–666. doi:10.1109/icstcc50638.2020.9259720
- Corke, P. I. (2017). *Robotics, Vision & Control: Fundamental Algorithms in MATLAB*, second edn. Berlin, Germany: Springer.
- Dao, P. N., Nguyen, H. Q., Nguyen, T. L., and Mai, X. S. (2021). Finite Horizon Robust Nonlinear Model Predictive Control for Wheeled mobile Robots. *Math. Probl. Eng.* 2021. doi:10.1155/2021/6611992

- Forbes, M. G., Patwardhan, R. S., Hamadah, H., and Gopaluni, R. B. (2015). Model Predictive Control in Industry: Challenges and Opportunities. *IFAC-PapersOnLine* 48, 531–538. doi:10.1016/j.ifacol.2015.09.022
- Franco, A., and Santos, V. (2019). “Short-term Path Planning with Multiple Moving Obstacle Avoidance Based on Adaptive Mpc,” in 2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC), Porto, Portugal, 24–26 April 2019 (IEEE), 1–7. doi:10.1109/icarsc.2019.8733653
- Frazzelle, C., Walker, I., AlAttar, A., and Kormushev, P. (2021). “Kinematic-model-free Control for Space Operations with Continuum Manipulators,” in 2021 IEEE Aerospace Conference (50100), Big Sky, MT, USA, 6–13 March 2021 (IEEE), 1–11. doi:10.1109/aero50100.2021.9438380
- Ji, J., Khajepour, A., Melek, W. W., and Huang, Y. (2016). Path Planning and Tracking for Vehicle Collision Avoidance Based on Model Predictive Control with Multiconstraints. *IEEE Trans. Vehicular Technology* 66, 952–964. doi:10.1109/LRA.2021.3083581
- Kamel, M., Alonso-Mora, J., Siegart, R., and Nieto, J. (2017). “Robust Collision Avoidance for Multiple Micro Aerial Vehicles Using Nonlinear Model Predictive Control,” in 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 Sept. 2017 (IEEE), 236–243. doi:10.1109/iros.2017.8202163
- Khosravi, M., Amirbande, M., Khaburi, D. A., Rivera, M., Riveros, J., Rodriguez, J., et al. (2019). Review of Model Predictive Control Strategies for Matrix Converters. *IET Power Electronics* 12, 3021–3032. doi:10.1049/iet-pel.2019.0212
- Kormushev, P., Demiris, Y., and Caldwell, D. G. (2015a). “Encoderless Position Control of a Two-Link Robot Manipulator,” in 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015 (IEEE), 943–949. doi:10.1109/icra.2015.7139290
- Kormushev, P., Demiris, Y., and Caldwell, D. G. (2015b). “Kinematic-free Position Control of a 2-dof Planar Robot Arm,” in 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, 28 Sept.–2 Oct. 2015 (IEEE), 5518–5525. doi:10.1109/iros.2015.7354159
- Krämer, M., Rösmann, C., Hoffmann, F., and Bertram, T. (2020). Model Predictive Control of a Collaborative Manipulator Considering Dynamic Obstacles. *Optim Control. Appl. Meth* 41, 1211–1232. doi:10.1002/oca.2599
- Li, H., and Yamamoto, S. (2016). “A Model-free Predictive Control Method Based on Polynomial Regression,” in 2016 SICE International Symposium on Control Systems (ISCS), Nagoya, Japan, 7–10 March 2016 (IEEE), 1–6. doi:10.1109/siceiscs.2016.7470167
- Li, J., Sun, J., Liu, L., and Xu, J. (2021). Model Predictive Control for the Tracking of Autonomous mobile Robot Combined with a Local Path Planning. *Meas. Control*. 54, 1319–1325. doi:10.1177/00202940211043070
- Li, W., and Xiong, R. (2019). Dynamical Obstacle Avoidance of Task- Constrained Mobile Manipulation Using Model Predictive Control. *IEEE Access* 7, 88301–88311. doi:10.1109/access.2019.2925428
- Nikou, A., Verginis, C., Heshmati-Alamdari, S., and Dimarogonas, D. V. (2017). “A Nonlinear Model Predictive Control Scheme for Cooperative Manipulation with Singularity and Collision Avoidance,” in 2017 25th Mediterranean Conference on Control and Automation (MED), Valletta, Malta, 3–6 July 2017 (IEEE), 707–712. doi:10.1109/med.2017.7984201
- Oleinikov, A., Kusdavletov, S., Shintemirov, A., and Rubagotti, M. (2021). Safety-aware Nonlinear Model Predictive Control for Physical Human-Robot Interaction. *IEEE Robotics Automation Lett.* 6, 5665–5672. doi:10.1109/lra.2021.3083581
- Pankert, J., and Hutter, M. (2020). Perceptive Model Predictive Control for Continuous mobile Manipulation. *IEEE Robot. Autom. Lett.* 5, 6177–6184. doi:10.1109/lra.2020.3010721
- Quirynen, R., Berntorp, K., Kambam, K., and Di Cairano, S. (2020). “Integrated Obstacle Detection and Avoidance in Motion Planning and Predictive Control of Autonomous Vehicles,” in 2020 American Control Conference (ACC), Denver, CO, USA, 1–3 July 2020 (IEEE), 1203–1208. doi:10.23919/acc45564.2020.9147820
- Saputra, R. P., Rakicevic, N., Chappell, D., Wang, K., and Kormushev, P. (2021). Hierarchical Decomposed-Objective Model Predictive Control for Autonomous Casualty Extraction. *IEEE Access* 9, 39656–39679. doi:10.1109/access.2021.3063782
- Song, Y., and Huh, K. (2021). Driving and Steering Collision Avoidance System of Autonomous Vehicle with Model Predictive Control Based on Non-convex Optimization. *Adv. Mech. Eng.* 13, 16878140211027669. doi:10.1177/16878140211027669
- Stenman, A. (1999). “Model-free Predictive Control,” in Proceedings of the 38th IEEE Conference on Decision and Control (Cat. No. 99CH36304), Phoenix, AZ, USA, 7–10 Dec. 1999 (IEEE), 3712–3717. doi:10.1109/CDC.1999.8279314
- Wang, L., Zhao, J., and Zhang, L. (2021). “Navdog: Robotic Navigation Guide Dog via Model Predictive Control and Human-Robot Modeling,” in Proceedings of the 36th Annual ACM Symposium on Applied Computing, 815–818.
- Yamamoto, S. (2014). A New Model-free Predictive Control Method Using Input and Output Data. *Amr* 1042, 182–187. doi:10.4028/www.scientific.net/AMR.1042.182
- Zanon, M., Boccia, A., Palma, V. G. S., Parenti, S., and Xausa, I. (2017). “Direct Optimal Control and Model Predictive Control,” in *Optimal Control: Novel Directions and Applications* (Springer), 263–382. doi:10.1007/978-3-319-60771-9_3

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher’s Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 AlAttar, Chappell and Kormushev. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.