

## Nonlinear Model Predictive Control of Rover Robotics System

Serdar Kalaycioglu\*, Anton de Ruiter

Department of Aerospace Engineering, Toronto Metropolitan University, Toronto, M5B 2K3, Canada

### ARTICLE INFO

Article history:

Received: 30 September, 2022

Accepted: 23 December, 2022

Online: 24 January, 2023

Keywords:

NMPC

Optimal Control

Multi Rover Control

### ABSTRACT

The paper presents two robust and efficient control algorithms based on (i) Optimal Control Allocation (OCA) and (ii) Nonlinear Model Predictive Control (NMPC). The robotics system consists of two rovers with mecanum wheels and mounted two 7-DOF arms carrying a common load. The overall system is an underdetermined one with non-holonomic constraints. The developed control algorithms focus on providing an optimal solution to the wheel and joint torque saturation problem, which is typically encountered while manipulating a large and heavy payload. The first control algorithm based on OCA minimizes a quadratic cost function consisting of robot joint and rover wheel torques, contact forces, and moments using only the current state values and the system dynamics. It is computationally very efficient. The NMPC algorithm minimizes a quadratic cost function which not only includes the current states but also the future state estimates, and the control inputs over a specified prediction horizon. The system consisting of multi-rover with a dual arm is highly non-linear. The linear MPC technique on which most of the previous studies relied is not adequate. On the other hand, the computational difficulties of a generic NMPC algorithm is remarkably high. In this paper, an elegant, discretized technique with exact realization is implemented to take into account the full non-linear model and yet provide a simple real-time solution satisfying a minimum performance index subject to constraints. The results show that the developed control algorithms OCA and NMPC work efficiently, and the minimum the contact moments and forces, and the joint torques are realized while two arms carry a common load and successfully track a reference end-effector trajectory. The results also indicate that although NMPC algorithm is computationally more involved, it provides superior results in reducing joint and wheel torques as well as contact moments and forces.

## 1. Introduction

This paper is an extension of the work originally presented in IEMTRONICS [1]. The Optimal Control Allocation algorithm (OCA) presented in the original work is further extended to accommodate a Nonlinear Model Predictive Control technique to increase performance of the approach.

There has been a significant interest in exploring complex environments using mobile rovers. Such rovers are commonly used in space exploration, construction, mining, and military.

Especially, there has been a considerable amount of interest in Space Robotics Exploration missions in the last two decades. Similar to on-orbit robotics missions (e.g., servicing, assembly,

and manufacturing), the future planetary exploration missions will also include tasks such as assembly of large space structures using multiple coordinating rovers and the rover-mounted robotics manipulators. Recently the Moon and Mars rover missions are the main target of various space agencies including NASA, Canadian Space Agency, ESA, JAXA, etc. Most of these space agencies in collaboration with space industries and research centers are heavily focusing on innovative rover technologies and designs. Autonomous rover motion control capability has been identified as one of the critical and enabling technology requirement for such systems. Although, there is a significant amount of research studies in the fields of control of single rover trajectory and force control of fixed-based arms, there are still major research challenges in the areas of load sharing multi-rovers and arms, particularly, real-time force and motion control when they are carrying a common load.

\* Corresponding Author: Serdar Kalaycioglu, TMU, Department of Aerospace Engineering, Toronto, Canada, email: [skalay@torontomu.ca](mailto:skalay@torontomu.ca)

The initial technological challenges that involved designing a mobile rover were related to its mechanics. These included the development of dynamic control systems and collision free trajectories.

In order to develop effective control systems for mobile rovers, a team led by Neculescu [2,3] studied the free and contact motion of the vehicles. They also developed methods to generate collision free trajectories and perform force control.

Motion control of rovers with nonholonomic constraints were studied using differential wheeled rovers in [4,5] These constraints exist if the constraints cannot be expressed in the form of time derivatives of a function consists of the generalized coordinates.

There have been extensive studies in control of systems with non-holonomic constraints. However, most of the cases, kinematic control is typically achieved by ignoring the dynamics when dealing with systems with non-holonomic constraints [6]. However, it has been shown that a mechanical system with these constraints were controlled in spite of its structure [7]. In addition, it has been shown that a non-holonomic system cannot be brought to a single equilibrium with a smooth time-invariant feedback [8].

In a study conducted in Kalaycioglu [9], a control technique with optimal force distribution for multiple robotic manipulators was demonstrated. However, it only involved two cooperating arms and did not include rovers.

The use of a Model Predictive Control (MPC) framework facilitates the optimization of a given performance index. It also allows for the analysis of the system constraints and dynamics [10–15]. One of the most challenging aspects of implementing a robust model of (MPC) is dealing with the various uncertainties that can impact its performance [16]. Due to the characteristics of the model's receding horizon, standard MPC can provide an adequate level of robustness [17].

Unfortunately, the literature has shown that standard MPC cannot provide an adequate performance in complex robotics systems [18]. To address this issue, various research studies have been conducted to develop novel MPC methods that can provide a robust and stable performance [19–23].

The scope and capabilities of Non-linear Model Predictive Control (NMPC) have significantly improved over the past few years. Due to the increasing number of tools that can be used to implement this type of model, the performance of this algorithm has been greatly improved. Some of these include the ability to perform fast gradient use and input parameterisation [24–27]. The application of NMPS for free-floating space manipulator are provided in [28–31].

The mechanics of wheeled locomotion have also attracted a lot of attention [32–37]. A number of studies have been conducted on the dynamics and kinematics of the mecanum wheel (a subcategory of omnidirectional wheel) [38–43].

There has been a significant amount of research on the various aspects of wheeled locomotion, but it is still not yet feasible to fully understand the mechanisms involved in the movement control of multiple rovers and mounted arms. For instance, the development of systems with multi- rovers with dual manipulators that can

perform real-time trajectories while manipulating a common load is still in its early stages.

This paper presents two robust and efficient control algorithms based on (i) Optimal Control Allocation (OCA) and (ii) Nonlinear Model Predictive Control (NMPC) for a rover robotics system with mecanum wheels when the two 7-DOF arms operating a common load. The system is an underdetermined one subject to non-holonomic constraints. The control algorithms focus on providing an optimal solution to the wheel and joint torque saturation problem, which is typically encountered while manipulating a large and heavy payload.

The first control algorithm based on OCA minimizes a quadratic cost function (a performance index) consisting of robot joint and rover wheel torques, contact forces, and moments using only the current state values and the system dynamics. It is computationally very efficient. The NMPC algorithm minimizes a quadratic cost function which not only includes the current states but also the future state estimates, and the control inputs over a specified prediction horizon.

The literature on the application of MPC for robotics is mainly focused on linear models. However, the multi-rover dual arm coordinating system is highly non-linear and MPC lacks robust applications in this area. In this paper, we present a novel NMPC discretized technique that incorporates the full non-linear characteristics of the multi-rover dual arm system.

This paper consists of four sections. The first section provides the mathematical formulations such as the kinematics and dynamics models of the total system including two  $n$ -degree redundant manipulators, two rovers and a common load. The second section presents two novel control algorithms based on optimal control allocation (OCA) and non-linear model predictive control (NMPC) which are formulated to minimize the wheel moments, the joint torques, and contact moments/forces. The third section provides the simulation results and discussion, and the fourth section provides some concluding remarks and recommendations for future work.

## 2. Theoretical Formulations

### 2.1. The Rover Robotics System

The system includes two mobile rovers with four mecanum wheels and two  $n$ -DOF redundant arms attached on the two rovers carrying a common load. Figure 1 shows an example of such a system with two rovers and two  $n$ -degree arms.

Table 1 contains the rover and robotics parameters utilized in the computer simulations.. The rotation angle  $\psi_i$  and the position vector  $\tilde{\mathbf{R}}_{ci}$ , provide the pose of the center of mass  $C_i$  of the  $i^{\text{th}}$  rover-in the inertial coordinate system, X, Y, Z. The coordinate axes  $x_{ci}$ ,  $y_{ci}$ ,  $z_{ci}$  attached to point  $C_i$  are obtained via a rotation around Z-axis with an angle of  $\psi_i$ .

The masses associated with the rovers and the wheels are given as  $m_{ci}$  and  $m_{wij}$ , respectively for the  $i^{\text{th}}$  rover and the  $j^{\text{th}}$  wheel,  $j=1...4$  and  $i=1,2$  for each rover. The distances between the wheel centers along the  $y_{ci}$  and  $x_{ci}$ -axes are denoted by  $2a$  and  $2b$ , respectively.

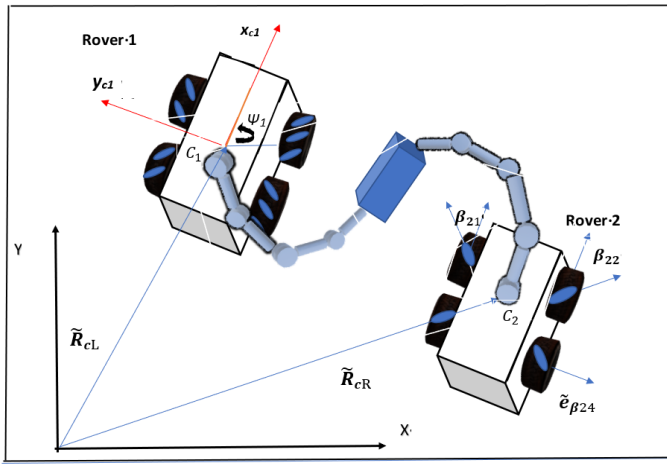


Figure 1: Description of the rover robotics system

The wheels have a radius of  $s$  and the angle of rotation, and the angular rate are denoted as  $\phi_{ij}$  and  $\omega_{ij}$ , respectively. The rollers are attached to the outer rims of the mecanum wheels as illustrated in Figure 1. The angle  $\beta_{ij}$  is defined as the angle between the axis of rotation of the roller and the  $x_{ci}$  of the  $j^{\text{th}}$  wheel of the  $i^{\text{th}}$  rover.

## 2.2. Model of Kinematics

$\tilde{V}_{mij}$ , the velocity vector of the center of the  $j^{\text{th}}$  wheel of the  $i^{\text{th}}$  rover can be determined by the following relationship:

$$\tilde{V}_{mij} = \tilde{V}_{ci} + \tilde{\Omega}_{ci} \times \tilde{r}_{wij} \quad (1)$$

$$\tilde{\Omega}_{ci} = \dot{\psi}_i \tilde{e}_z \quad (2)$$

where  $\tilde{V}_{ci}$  is the velocity of the mass center of the rover,  $\tilde{\Omega}_{ci}$  is the angular velocity vector of the rover and  $\tilde{e}_z$  is a unit vector both along the  $z_{ci}$  axis while  $\tilde{r}_{wij}$  is the position vector from the rover's mass center to the wheel center.

The velocity vector  $\tilde{V}_{pij}$ , representing the velocity of a point  $P$  located at the roller center can be expressed as

$$\tilde{V}_{pij} = \tilde{V}_{mij} + \tilde{\omega}_{ij} \times \tilde{\rho}_{ij} \quad (3)$$

where  $\tilde{\rho}_{ij}$  is the position vector from the wheel's center to the point  $P$ , the roller center.

If the rollers do not slip,  $\tilde{V}_{pij}$  does not have a component in the direction of the axis of roller rotation  $\tilde{e}_{\beta ij}$ , and can be expressed as

$$\tilde{V}_{pij} \cdot \tilde{e}_{\beta ij} = 0 \quad (4)$$

where  $\tilde{e}_{\beta ij}$  is a unit vector along the roller's axis of rotation. After carrying out some algebraic manipulations using (3) and (4), one can write the following expressions:

$$\begin{aligned} \tilde{V}_{mij} \cdot \tilde{e}_{\beta ij} + (\tilde{\omega}_{ij} \times \tilde{\rho}_{ij}) \cdot \tilde{e}_{\beta ij} &= 0 \\ (\tilde{\omega}_{ij} \times \tilde{\rho}_{ij}) &= -\tilde{\omega}_{ij} s \tilde{e}_{xi} \end{aligned}$$

$$\tilde{V}_{mij} \cdot \tilde{e}_{\beta ij} = \tilde{\omega}_{ij} s (\tilde{e}_{xi} \cdot \tilde{e}_{\beta ij}) \quad (5)$$

where  $s$  is the radius of the wheel and  $\tilde{e}_{xi}$  is a unit vector in the direction of the  $x_{ci}$  axis.

Furthermore, rewriting the equations of constraints by utilizing (1) and (5), one can obtain the following relationships:

$$\begin{aligned} \tilde{V}_{ci} \cdot \tilde{e}_{\beta ij} + (\tilde{\Omega}_{ci} \times \tilde{r}_{wij}) \cdot \tilde{e}_{\beta ij} &= \tilde{\omega}_{ij} s (\tilde{e}_{xi} \cdot \tilde{e}_{\beta ij}) \\ \tilde{V}_{ci} \cdot \tilde{e}_{\beta ij} + (\tilde{r}_{wij} \times \tilde{e}_{\beta ij}) \cdot \tilde{\Omega}_{ci} &= \tilde{\omega}_{ij} s \cos(\beta_{ij}) \end{aligned} \quad (6)$$

where  $\beta_{ij}$  is defined as the angle between the two unit vectors  $\tilde{e}_{xi}$  and  $\tilde{e}_{\beta ij}$

$$\begin{aligned} e_{\beta i1}^T &= [\cos(\beta_{i1}), -\sin(\beta_{i1}), 0] \\ e_{\beta i2}^T &= [\cos(\beta_{i2}), \sin(\beta_{i2}), 0] \\ e_{\beta i3}^T &= [\cos(\beta_{i3}), \sin(\beta_{i3}), 0] \\ e_{\beta i4}^T &= [\cos(\beta_{i4}), -\sin(\beta_{i4}), 0] \\ \tilde{r}_{wi1}^T &= [a, b, 0] \\ \tilde{r}_{wi2}^T &= [a, -b, 0] \\ \tilde{r}_{wi3}^T &= [-a, b, 0] \\ \tilde{r}_{wi4}^T &= [-a, -b, 0] \end{aligned} \quad (7)$$

One can obtain the following expressions by plugging (7) into (6) and substituting  $45^\circ$  for  $\beta_{ij}$ :

$$\begin{aligned} \tilde{V}_{ci} &= \begin{bmatrix} V_{cix} \\ V_{ciy} \\ V_{ciz} \end{bmatrix} = \begin{bmatrix} s(\omega_{i1} + \omega_{i2})/2 \\ s(\omega_{i3} - \omega_{i1})/2 \\ 0 \end{bmatrix} \\ \tilde{\Omega}_{ci} &= \begin{bmatrix} \Omega_{cix} \\ \Omega_{ciy} \\ \Omega_{ciz} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ s(\omega_{i3} - \omega_{i1})/(2(a+b)) \end{bmatrix} \\ \omega_{i4} &= \omega_{i1} + \omega_{i2} - \omega_{i3} \end{aligned} \quad (8)$$

The following rotational matrix represents the rotation between the inertial and the rover body axes:

$$\underline{\Psi}_{zi} = \begin{bmatrix} \cos\psi_i & -\sin\psi_i & 0 \\ \sin\psi_i & \cos\psi_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (9)$$

Homogeneous transformation matrix  $\underline{T}_f^g$  which transforms the coordinates between frame-g and frame-f on the robot arm can be obtained by Denavit-Hartenberg (D-H) convention as follows.

$$\underline{T}_f^g = \underline{A}_{f+1} \underline{A}_{f+2} \dots \underline{A}_{g-1} \underline{A}_g \quad f < g$$

$$\underline{A}_f = \begin{bmatrix} \cos \theta_{fi} & -\sin \theta_{fi} \cos \alpha_{fi} & \sin \theta_{fi} \sin \alpha_{fi} & a_{fi} \cos \theta_{fi} \\ \sin \theta_{fi} & \cos \theta_{fi} \cos \alpha_{fi} & -\cos \theta_{fi} \sin \alpha_{fi} & a_{fi} \sin \theta_{fi} \\ 0 & \sin \alpha_{fi} & \cos \alpha_{fi} & d_{fi} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (10)$$

where  $\theta_{fi}, \alpha_{fi}, d_{fi}, a_{fi}$  are the parameters related to joint-f and link-f on the  $i^{\text{th}}$  arm, namely  $d_{fi}$  is the offset,  $a_{fi}$  is the  $f^{\text{th}}$  link-length,  $\theta_{fi}$  is the joint angle and  $\alpha_{fi}$  is the twist as defined in DH convention.

The following expressions can be used to obtain the Jacobian matrices and the first-time derivatives of these matrices associated with the rover's center and any arbitrary point-k on the arm:

$$\left(\underline{J}_c^k\right)_i = \begin{bmatrix} \tilde{e}_z & \tilde{e}_1 & \dots & \tilde{e}_7 \\ \tilde{e}_z \times \tilde{r}_{ck} & \tilde{e}_1 \times \tilde{r}_{1k} & \dots & \tilde{e}_7 \times \tilde{r}_{7k} \end{bmatrix} \quad (11)$$

$$\left(\underline{\dot{J}}_c^k\right)_i = \begin{bmatrix} \tilde{e}_z & \tilde{e}_1 & \dots & \tilde{e}_7 \\ \tilde{e}_z \times (\tilde{\Omega}_{ci} \times \tilde{r}_{ck}) & \tilde{e}_1 \times (\dot{\theta}_1 \tilde{e}_1 \times \tilde{r}_{1k}) & \dots & \tilde{e}_7 \times (\dot{\theta}_7 \tilde{e}_7 \times \tilde{r}_{7k}) \end{bmatrix} \quad (12)$$

Where  $\tilde{e}_i$  is the unit vector along the  $i^{\text{th}}$  joint rotation axis,  $\tilde{e}_z$  is the unit vector along  $\tilde{\Omega}_{ci}$ , and  $\tilde{r}_{ik}, \tilde{r}_{ck}$  are the position vectors from  $i^{\text{th}}$  joint and the rover's center to the point k, respectively.

The linear and angular velocities and accelerations of point k on the  $i^{\text{th}}$  rover arm can be calculated as follows:

$$\begin{bmatrix} \tilde{\Omega}_k \\ \tilde{V}_k \end{bmatrix}^i = \left(\underline{\dot{J}}_c^k\right)_i \begin{bmatrix} \tilde{\Omega}_{ci} \\ \dot{\theta}_i \end{bmatrix} + \begin{bmatrix} 0 \\ \tilde{V}_{ci} \end{bmatrix} \quad (13)$$

$$\begin{bmatrix} \dot{\tilde{\Omega}}_k \\ \dot{\tilde{V}}_k \end{bmatrix}^i = \left(\underline{\dot{J}}_c^k\right)_i \begin{bmatrix} \dot{\tilde{\Omega}}_{ci} \\ \ddot{\theta}_i \end{bmatrix} + \left(\underline{\dot{J}}_c^k\right)_i \begin{bmatrix} \tilde{\Omega}_{ci} \\ \dot{\theta}_i \end{bmatrix} + \begin{bmatrix} 0 \\ \dot{\tilde{V}}_{ci} \end{bmatrix} \quad (14)$$

where  $\begin{bmatrix} \tilde{\Omega}_k \\ \tilde{V}_k \end{bmatrix}^i$  is a vector consisting of the angular and linear velocity vectors of the point k on the  $i^{\text{th}}$  arm, respectively while  $\dot{\tilde{\theta}}_i^T = [\dot{\theta}_{i1}, \dot{\theta}_{i2}, \dot{\theta}_{i3}, \dots, \dot{\theta}_{in}]$  is a vector consists of the  $i^{\text{th}}$  rover-arm joint angular rates.

### 2.3. Model of Dynamics

The dynamics equations of motions of the system is derived using the Lagrangian formulation. The total kinetic energy  $T_t$  consists of two parts, the rotational and translational kinetic energies of the robotics arms and the rovers.

$$T_t = T_{tr} + T_{rt} \quad (15)$$

The angular and translational velocities of the rovers as well as that of the robot links' center of mass can be calculated using

(14) and (8). Then, the total kinetic energy of the system can be obtained using (15).

The dynamics equations of motion can be obtained using the following Lagrangian formulation:

$$\frac{d}{dt} \left( \frac{\partial T_t}{\partial \dot{q}_h} \right) - \frac{\partial T_t}{\partial q_h} = Q_h, \quad h = 1, \dots, 2m \quad (16)$$

where  $q_h$  and  $Q_h$  are the generalized coordinates and forces, respectively and

$$q^T = [\phi_{11}, \phi_{12}, \phi_{13}, \phi_{21}, \phi_{22}, \phi_{23}, \theta_{11}, \dots, \theta_{1n}, \theta_{21}, \dots, \theta_{2n}]$$

and  $m=(n+3)$ ,  $n$  represents the total number of degrees of freedom of the robotics arms.

Applying (16), the dynamics equations of motions for both rovers and the arms can be written in the following form:

$$\begin{bmatrix} G_{WL} & G_{WLR} & G_{W\theta L} & G_{W\theta R} \\ G_{WLR}^T & G_{WR} & G_{W\theta L} & G_{W\theta R} \\ G_{W\theta L}^T & G_{W\theta L} & G_{\theta L} & G_{\theta LR} \\ G_{W\theta R}^T & G_{W\theta R} & G_{\theta LR} & G_{\theta R} \end{bmatrix} \begin{bmatrix} \ddot{\Phi}_L \\ \ddot{\Phi}_R \\ \ddot{\theta}_L \\ \ddot{\theta}_R \end{bmatrix} + \begin{bmatrix} \tilde{c}_L \\ \tilde{c}_R \\ \tilde{c}_{\theta L} \\ \tilde{c}_{\theta R} \end{bmatrix} = \begin{bmatrix} \tilde{M}_L \\ \tilde{M}_R \\ \tilde{\tau}_{\theta L} \\ \tilde{\tau}_{\theta R} \end{bmatrix} \quad (17)$$

where  $G$  is the mass / inertia matrix (a positive definite matrix) and,  $\ddot{\Phi}_L, \ddot{\Phi}_R$  are the wheels' angular accelerations for the two rovers  $i=L$  and  $R$ , and  $\ddot{\theta}_L, \ddot{\theta}_R$  are the joint rotational accelerations for the two manipulators,  $i=L$  and  $R$ , respectively. The indices L and R are referred to the first and second rover and robotics arm, respectively.

The non-linear Coriolis and centrifugal terms are represented by  $\tilde{c}_{L^L}, \tilde{c}_R, \tilde{c}_{\theta L}$ , and  $\tilde{c}_{\theta R}$ . and  $\tilde{\tau}_{\theta L}, \tilde{\tau}_{\theta R}$  are the joint control torques for the two robot manipulators. Finally,  $\tilde{M}_L, \tilde{M}_R$  are the wheel control moments for the two rovers.

$\dot{\tilde{\Phi}}_i^T = [\omega_{i1}, \omega_{i2}, \omega_{i3}]$  includes the wheels angular rates of the  $i^{\text{th}}$  rover,  $i=L$  and  $R$  for two rovers. If there is no slip, the fourth wheel angular rate can be calculated using (8).

### 2.4. Optimal Control Allocation (OCA) Technique

The robotics system composed of two rovers and two redundant arms is an undetermined because of the excessive number of sensors and actuators used to control the motions of the links and rovers.

A novel two-stage optimal control technique is derived in this section and the control system block diagram is provided in Figure 2a.

The first stage of the diagram generates the reference trajectories for the end-effectors corresponding to a given payload trajectory. The Impedance control equations representing this first stage are provided in (18). These equations are developed in [2].

$$\ddot{X}_i = \underline{M}_i^{-1} \underline{B} \{ \dot{X}_{di} - \dot{X}_i \} + \underline{M}_i^{-1} \underline{K} \{ X_{di} - X_i \}, \quad i = L, R$$

(18)

where,  $M_i, K, B$ , are 6x6 positive definite matrices and are chosen in accordance with the tracking performance requirements.  $\tilde{X}_i$ . (i varies between L and R for each arm) are the end-effector trajectories while  $\tilde{X}_{di}$ . correspond to the reference trajectories of the attachment points on the common load.

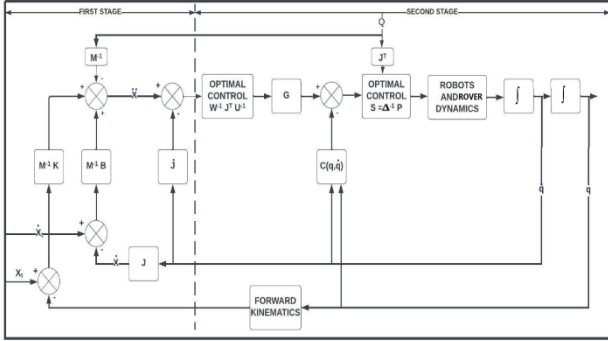


Figure 2: a. Optimal control system block diagram – Two Stage Control

The expressions for  $\tilde{X}_i$  and  $\dot{\tilde{X}}_i$  can be written as follows:

$$\tilde{X}_i = \begin{bmatrix} \tilde{\Omega}_k \\ \tilde{V}_k \end{bmatrix}^i \quad (19)$$

$$\dot{\tilde{X}}_i = \begin{bmatrix} \dot{\tilde{\Omega}}_k \\ \dot{\tilde{V}}_k \end{bmatrix}^i \quad (20)$$

where  $k$  point is the end-effector for the  $i^{\text{th}}$  arm. Performing the least-square minimization of joint rates, the inverse kinematics of the robotics system can be solved as the following:

$$\begin{bmatrix} \tilde{\Omega}_{ci} \\ \dot{\tilde{\theta}}_i \end{bmatrix} = \underline{W}_i^{-1} \left( \underline{J}_c^k \right)_i^T \underline{U}_i^{-1} \dot{\tilde{X}}_i \quad (21)$$

$$\begin{bmatrix} \dot{\tilde{\Omega}}_{ci} \\ \ddot{\tilde{\theta}}_i \end{bmatrix} = \underline{W}_i^{-1} \left( \underline{J}_c^k \right)_i^T \underline{U}_i^{-1} \left\{ \dot{\tilde{X}}_i - \left( \underline{J}_c^k \right)_i \begin{bmatrix} \tilde{\Omega}_{ci} \\ \dot{\tilde{\theta}}_i \end{bmatrix} \right\} \quad (22)$$

$$\underline{U}_i = \left( \underline{J}_c^k \right)_i \underline{W}_i^{-1} \left( \underline{J}_c^k \right)_i^T \quad (23)$$

where  $\underline{W}_i$  is a square positive definite weighting matrix with the dimensions of  $(n+3)$  by  $(n+3)$ .

The second stage in the block diagram is predicated on optimal control allocation (OCA). The mathematical model is provided below.

The performance index (a cost function)  $C$  is formulated to minimize the wheel moments  $\tilde{M}_L, \tilde{M}_R$  and the joint torques  $\tilde{\tau}_{\theta_L}, \tilde{\tau}_{\theta_R}$ , and the contact moments and forces  $\tilde{N}_i$  and  $\tilde{F}_i$  applied by the end-effectors on the common load

The performance index  $C$  can be expressed as:

$$C = \frac{1}{2} \tilde{S}^T \underline{H} \tilde{S} + \tilde{\lambda}^T \tilde{E} \quad (24)$$

$\underline{H}$  is a  $(2n+18, 2n+18)$  positive definite weighting matrix,  $\tilde{\lambda}$  is the  $((2n+12), 1)$  Lagrangian multiplier and  $\tilde{E}$  vector includes the equations of constraints and can be calculated as shown in (26).

The  $\tilde{S}$  vector contains the contact forces / moments, the wheel moments as well as the joint torques for the two arms and rovers as described below:

$$\tilde{S}^T = [\tilde{Q}_L, \tilde{Q}_R, \tilde{M}_L, \tilde{M}_R, \tilde{\tau}_{\theta_L}, \tilde{\tau}_{\theta_R}]$$

$$\tilde{Q}_i = \begin{bmatrix} \tilde{N}_i \\ \tilde{F}_i \end{bmatrix} \quad (25)$$

The  $\tilde{E}$  vector is provided below:

$$\tilde{E} = \begin{bmatrix} m_t \ddot{x}_t - \tilde{F}_L - \tilde{F}_R \\ [L_t \tilde{\Omega}_t + \tilde{\Omega}_t \times L_t \tilde{\Omega}_t] - [\tilde{N}_L + \tilde{N}_R - \tilde{d}_L \times \tilde{F}_L - \tilde{d}_R \times \tilde{F}_R] \\ G_{W_L} \quad G_{W_{LR}} \quad G_{W_{\theta_L}} \quad G_{W_{\theta_R}} \\ G_{W_{LR}}^T \quad G_{W_R} \quad G_{W_{\theta_L}} \quad G_{W_{\theta_R}} \\ G_{W_{\theta_L}}^T \quad G_{W_{\theta_L}} \quad G_{\theta_L} \quad G_{\theta_{LR}} \\ G_{W_{\theta_R}}^T \quad G_{W_{\theta_R}} \quad G_{\theta_{LR}} \quad G_{\theta_R} \end{bmatrix} \begin{bmatrix} \ddot{\Phi}_L \\ \ddot{\Phi}_R \\ \ddot{\theta}_L \\ \ddot{\theta}_R \end{bmatrix} - \begin{bmatrix} \tilde{c}_L \\ \tilde{c}_R \\ \tilde{c}_{\theta_L} \\ \tilde{c}_{\theta_R} \end{bmatrix} - \begin{bmatrix} \tilde{M}_L \\ \tilde{M}_R \\ \tilde{\tau}_{\theta_L} \\ \tilde{\tau}_{\theta_R} \end{bmatrix} \quad (26)$$

where  $\ddot{x}_t, m_t$  are the translational acceleration and the mass of the common load, respectively and.  $\tilde{d}_i^T = (x_i, y_i, z_i)$  is the position vector measured from the  $i^{\text{th}}$  arm's contact point to the load's mass center, while  $\tilde{\Omega}_t$  and  $L_t$  are the angular rate and the inertia matrix of the common load around its center of mass.

Once can minimize the performance index  $C$  by taking the derivative of  $C$  with respect to  $\tilde{\lambda}_i$  and  $\tilde{S}$  to obtain the minimum norm of wheel moments, joint torques, as well as the contact moments /force exerted by the end-effectors on the common load.

$$\frac{\partial C}{\partial \tilde{S}} = \tilde{0} \quad (27)$$

and

$$\frac{\partial C}{\partial \tilde{\lambda}} = \tilde{0} \quad (28)$$

One can obtain the minimum norm of  $\tilde{\mathcal{S}}$  containing the wheel moments, joint torques, as well as the contact force and moments by making use of the equations (27) and (28).

$$\tilde{\mathcal{S}} = \underline{\Delta}^{-1} \tilde{\mathcal{P}} \quad (29)$$

where  $\underline{\Delta}$  is a  $((2n + 18), (2n + 18))$  square matrix and  $\underline{\Delta}$  and  $\tilde{\mathcal{P}}$  are presented as follows:

$$\underline{\Delta} = \begin{bmatrix} \underline{H}_{NL} & \underline{0} & -\underline{H}_{NR} & \underline{0} & -\underline{H}_{rL} (J_c^k)_L^T & \underline{W}_{rR} (J_c^k)_R^T \\ (\underline{D}_L - \underline{D}_R) \underline{H}_{NL} & \underline{H}_{FL} & \underline{0} & -\underline{H}_{FR} & (\underline{D}_R - \underline{D}_L - \underline{1}) (J_c^k)_L^T & \underline{W}_{rR} (J_c^k)_R^T \\ \underline{0} & \underline{1} & \underline{0} & \underline{1} & \underline{0} & \underline{0} \\ \underline{1} & -\underline{D}_L & \underline{1} & \underline{D}_R & \underline{0} & \underline{0} \\ (J_c^k)_{L1}^T & (J_c^k)_{L2}^T & \underline{0} & \underline{0} & \underline{1} & \underline{0} \\ \underline{0} & \underline{0} & (J_c^k)_{R1}^T & (J_c^k)_{R2}^T & \underline{0} & \underline{1} \end{bmatrix} \quad (30)$$

$$\tilde{\mathcal{P}} = \begin{bmatrix} \underline{0} \\ \underline{0} \\ \underline{0} \\ \underline{0} \\ m_t \ddot{x}_t \\ \underline{L}_t \dot{\tilde{\Omega}}_t + \tilde{\Omega}_t \times \underline{L}_t \tilde{\Omega}_t \\ \begin{bmatrix} G_{WL} & G_{WLR} & G_{W\theta L} & G_{W\theta R} \\ G_{WL}^T & G_{WR} & G_{W\theta L} & G_{W\theta R} \\ G_{W\theta L}^T & G_{W\theta L} & G_{\theta L} & G_{\theta LR} \\ G_{W\theta R}^T & G_{W\theta R} & G_{\theta LR}^T & G_{\theta R} \end{bmatrix} \begin{bmatrix} \ddot{\Phi}_L \\ \ddot{\Phi}_R \\ \ddot{\Theta}_L \\ \ddot{\Theta}_R \end{bmatrix} \end{bmatrix} + \begin{bmatrix} \tilde{c}_L \\ \tilde{c}_R \\ \tilde{c}_{\theta L} \\ \tilde{c}_{\theta R} \end{bmatrix} \quad (31)$$

$$\underline{D}_i = \begin{bmatrix} 0 & -z_i & y_i \\ z_i & 0 & -x_i \\ -y_i & x_i & 0 \end{bmatrix} \quad (32)$$

### 2.5. Non-linear Model Predictive Control (NMPC) Technique

The control block diagram of the NMPC is illustrated in Figure.2b. It replaces the second stage of the model in Figure.2a. The reference trajectory shown in this diagram is the output of the first stage, i.e., the impedance control trajectory generation. However, in this case, the future state estimates are also taken into account to estimate the future reference trajectory values.

A robust NMPC algorithm is implemented by optimizing a performance index of the system which considers the predictions of the output signal and the constraints on the states, outputs and inputs as illustrated in Figure 2b.

The main difference between the Optimal Control Allocation (OCA) and the Non-linear Model Predictive Control (NMPC) is that the latter utilizes a model to predict and control future behavior, while the former only takes into account the current and the past.

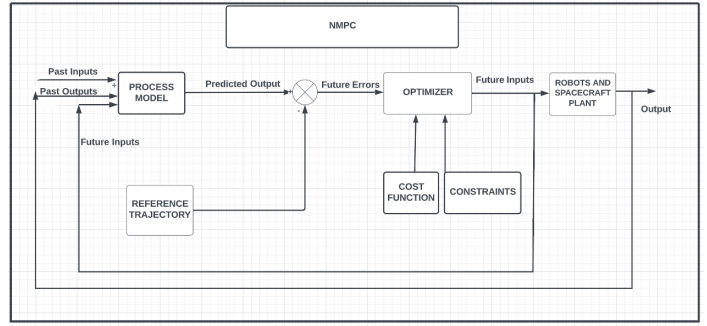


Figure 2: b Nonlinear Model Predictive Control (NMPC) block diagram

The optimization process carried out through the NMPC algorithm is performed at each control interval to predict the system's future behavior. It involves implementing various optimization problems related to the cost functions and constraints. The cost function is a type of scalar which needs to be minimized at intervals to assess the system's performance.

Besides the cost functions, the system also has to perform under various constraints to check its performance. These include the plant output and states. The modified states are adjusted depending on the constraints that are applied to the system.

The conventional MPC formulation for the multi-rover nonlinear system can be written as:

$$C = \int_0^{T_p} \left[ (\tilde{\mathcal{Y}}(t) - \tilde{\mathcal{Y}}_r(t))^T \underline{K} (\tilde{\mathcal{Y}}(t) - \tilde{\mathcal{Y}}_r(t)) + \tilde{\mathcal{S}}^T(t) \underline{H} \tilde{\mathcal{S}}(t) \right] dt$$

subject to:

$$\begin{aligned} \dot{\tilde{\mathcal{Y}}} &= \tilde{\mathcal{G}}(\tilde{\mathcal{Y}}) + \underline{L} \tilde{\mathcal{S}} \\ \tilde{\mathcal{Z}} &= \tilde{\mathcal{G}}_z(\tilde{\mathcal{Y}}) + \underline{H} \tilde{\mathcal{S}} \\ \tilde{\mathcal{Y}}(0) &= \tilde{\mathcal{Y}}(t_0) \end{aligned} \quad (33)$$

where  $T_p$  is the prediction horizon;  $\underline{K}$  and  $\underline{H}$  are  $(2(2n+6) \times 2(2n+6))$  and  $((2n+18) \times (2n+18))$  positive definite square weighting matrices, respectively;  $\tilde{\mathcal{G}}(\tilde{\mathcal{Y}})$ ,  $\tilde{\mathcal{G}}_z(\tilde{\mathcal{Y}})$ ,  $\underline{L}$ ,  $\underline{H}$  are part of the nonlinear system equations.

Also,  $\tilde{\mathcal{Y}}(t)^T = [q^T, \dot{q}^T]$ , is a  $(1 \times 2(6+2n))$  state vector,  $q^T$  vector is previously defined in Eq.(16), and  $\tilde{\mathcal{Y}}_r(t)$  is the reference / desired states.

The non-linear system can now be described with an exact quasi linear parameter varying realization:

$$\begin{aligned} \tilde{\mathcal{Y}}(k_t + 1) &= \hat{\underline{A}}(\hat{\mathcal{g}}(k_t)) \tilde{\mathcal{Y}}(k_t) + \hat{\underline{B}}(\hat{\mathcal{g}}(k_t)) \tilde{\mathcal{S}}(k_t) \\ \tilde{\mathcal{Z}}(k_t) &= \hat{\underline{C}}(\hat{\mathcal{g}}(k_t)) \tilde{\mathcal{Y}}(k_t) + \hat{\underline{D}}(\hat{\mathcal{g}}(k_t)) \tilde{\mathcal{S}}(k_t) \\ \hat{\mathcal{g}}(k_t) &= \mathbf{f}_g(\tilde{\mathcal{Y}}(k_t)) \end{aligned} \quad (34)$$

where  $k_t$  is the sampling instant and  $\tilde{\mathcal{Z}}(k_t)$  is a vector of the measured outputs at instant  $k_t$ .

The NMPC is employed at each sampling instant  $k_t$ . and the discrete states  $\tilde{\mathcal{Y}}(k_t)$  and control inputs  $\tilde{\mathcal{S}}(k_t)$  are obtained

minimizing the following performance index i.e., the Cost Function:

$$C = \frac{1}{2} \sum_{j=1}^{N_p} \left[ \begin{array}{c} (\tilde{\mathbf{y}}(k_t, +j) - \tilde{\mathbf{y}}_r(k_t, +j))^T \underline{K} (\tilde{\mathbf{y}}(k_t, +j) - \tilde{\mathbf{y}}_r(k_t, +j)) \\ + \tilde{\mathbf{S}}(k_t, +j - 1)^T \underline{H} \tilde{\mathbf{S}}(k_t, +j - 1) \end{array} \right]$$

subject to

$$\tilde{\mathbf{y}}(k_t + j + 1) = \underline{\mathbf{A}}(\hat{\mathbf{g}}(k_t + j))\tilde{\mathbf{y}}(k_t + j) + \underline{\mathbf{B}}(\hat{\mathbf{g}}(k_t + j)) \tilde{\mathbf{S}}(k_t + j)$$

$$\tilde{\mathbf{z}}(k_t + j) = \underline{\mathbf{C}}(\hat{\mathbf{g}}(k_t + j))\tilde{\mathbf{y}}(k_t + j) + \underline{\mathbf{D}}(\hat{\mathbf{g}}(k_t + j)) \tilde{\mathbf{S}}(k_t + j) \tag{35}$$

### 3. Computer Simulation Results and Discussion

The results of the computer simulations and their discussions are presented in this section. First, a prescribed trajectory for the common payload's center mass is generated. Then, the desired (reference) trajectories for the two end-effectors are obtained using a method known as the impedance control technique (shown as the first stage in the block diagram).

The goal of the simulation is to obtain the minimum joint and rover wheel torques and contact forces while simultaneously tracking the desired end-effector pose using the developed two different control algorithms (i) OCA and (ii) NMPC.

The parameters for the robotic arms and the rovers employed in the computer simulations are presented in Table 1. A mini version of the SSRMS is utilized.

Table 1: The System Parameters Utilized in the Computer Simulation

Description of Hardware Configuration Items	Dimensions (m)	Mass (kg)
Rovers- (#1 and #2)	(0.5x0.5x0.3)	40
Common Load	(0.4x1x0.4)	10
Link #1	(0.1x0.1x0.1)	1
Link #2	(0.1x0.1x0.1)	1
Link #3	(1x0.1x0.1)	3
Link #4	(1x0.1x0.1)	5
Link #5	(0.1x0.1x0.1)	3
Link #6	(0.1x0.1x0.1)	1
Link #7	(1x0.1x0.1)	3

The desired trajectories for the rotational and translational motions of the common load are presented with time in Figure 3.

The minimum norm of the contact moments /forces, the joint torques, as well as the control forces and moments on Rovers 1 and 2 are plotted in Figure 4a-m using OCA and NMPC algorithms. The non-optimum joint torques (in blue), the joint torques realized by application of OCA algorithm (in red) and by NMPC algorithm (in yellow) are plotted for comparison purposes. The comparison

of the plots illustrates that the NMPC is superior and then followed by OCA.

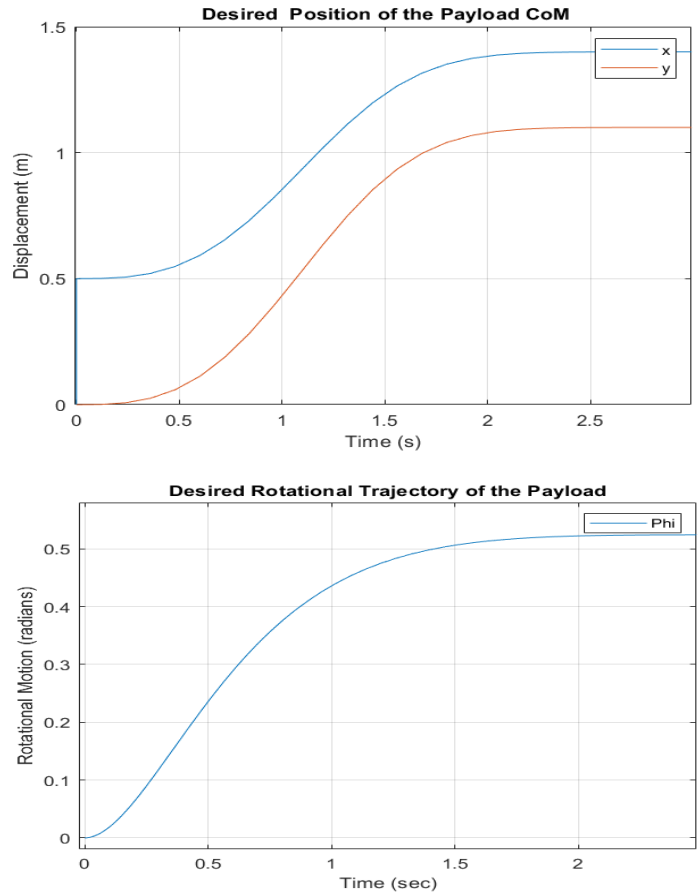


Figure 3: Variation of the reference trajectory for the common load

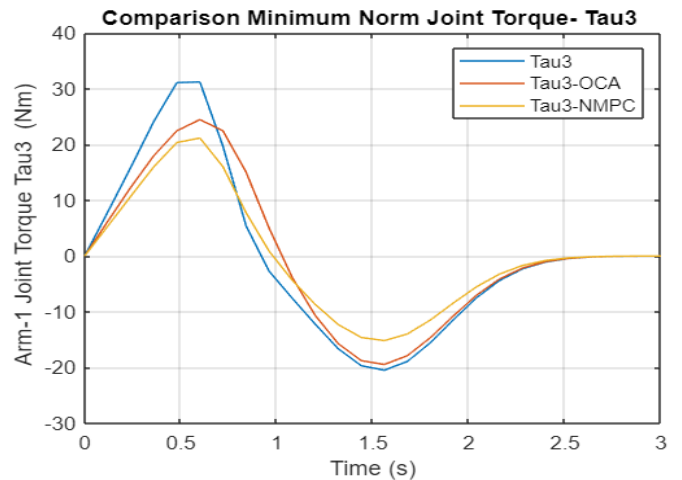


Figure 4: a Variation of the Joint 3 Torque obtained by Non-optimal, OCA, NMPC Algorithms (Arm 1)

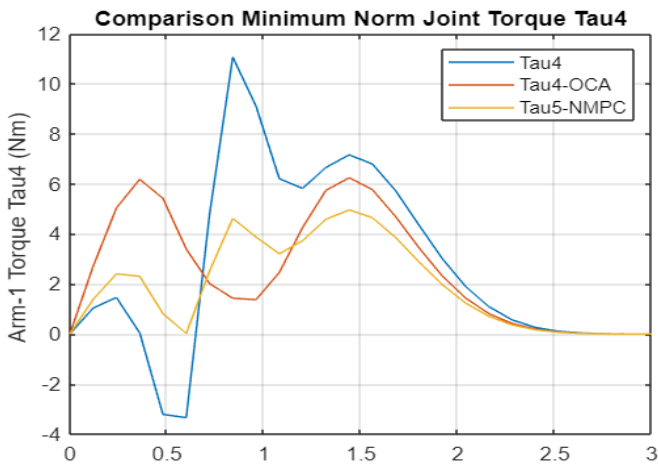


Figure 4: b-Variation of the Joint 4 Torque obtained by Non-optimal, OCA, NMPC Algorithms (Arm 1)

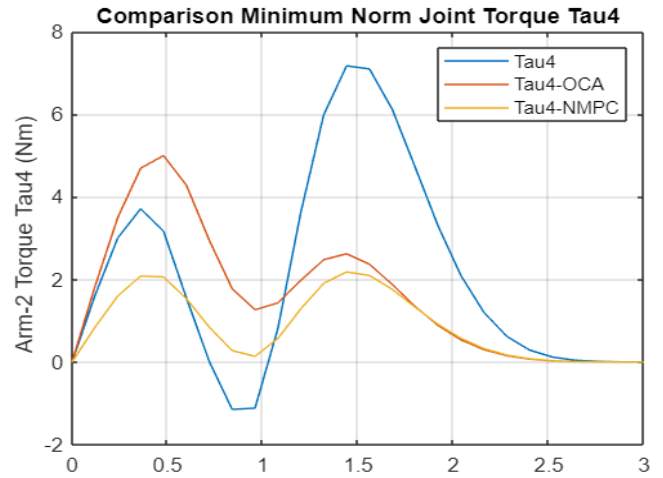


Figure 4: e Variation of the Joint 4 Torque obtained by Non-optimal, OCA, NMPC Algorithms (Arm 2)

Comparison Minimum Norm Joint Torque Tau5 - In Orbit Plan

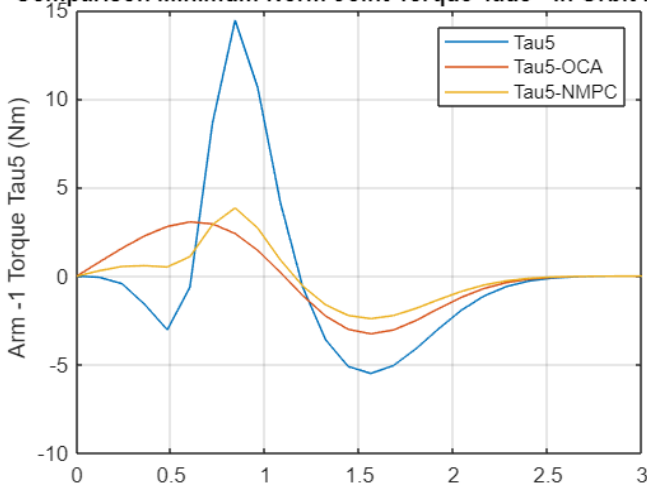


Figure 4: c Variation of the Joint 5 Torque obtained by Non-optimal, OCA, NMPC Algorithms (Arm 1)

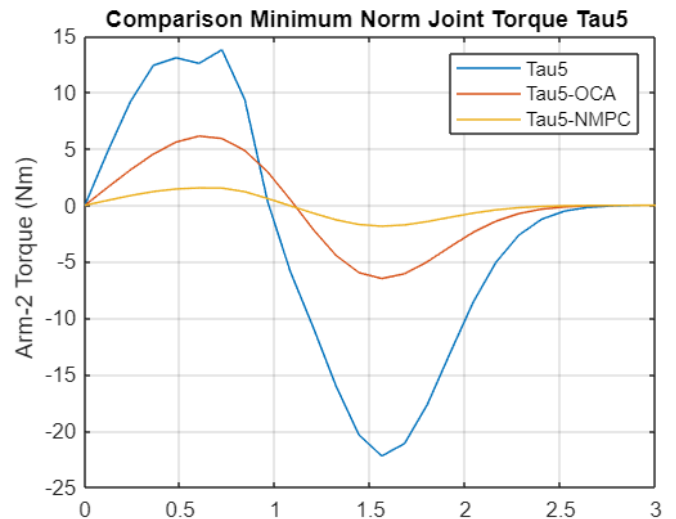


Figure 4: f Variation of the Joint 5 Torque obtained by Non-optimal, OCA, NMPC Algorithms (Arm 2)

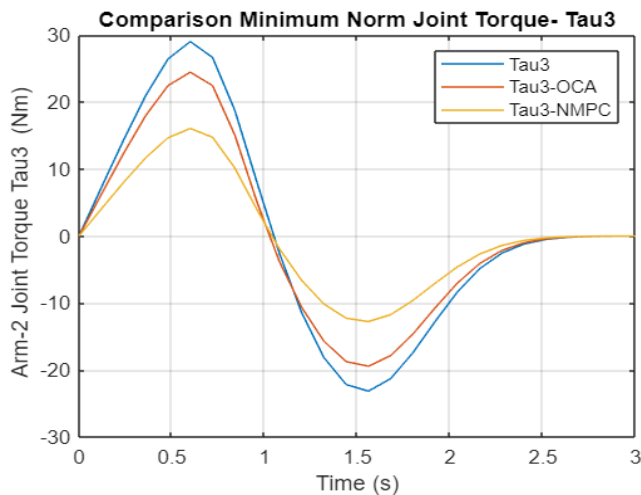


Figure 4: d Variation of the Joint 3 Torque obtained by Non-optimal, OCA, NMPC Algorithms (Arm 2)

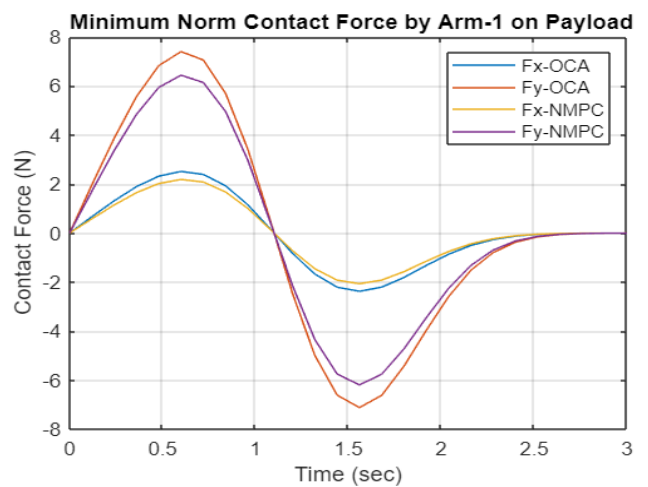


Figure 4: g Variation of the Contact Forces on Payload by Non-optimal, OCA, NMPC Algorithms (Arm 1)



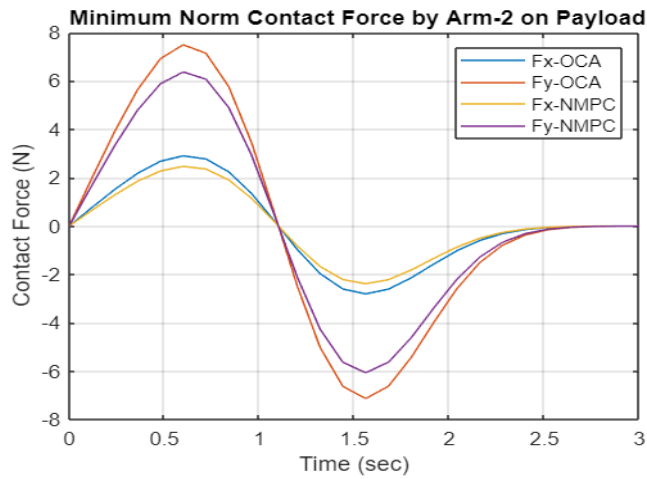


Figure 4: h Variation of the Contact Forces on Payload by Non-optimal, OCA, NMPC Algorithms (Arm 2)

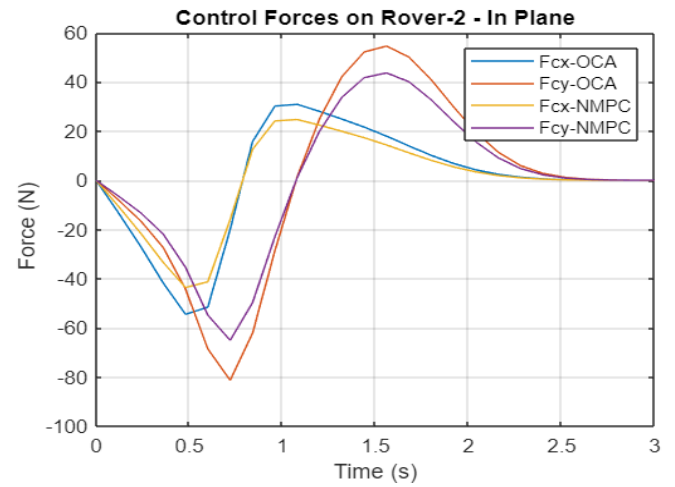


Figure 4: k Variation of the Control Forces on Rover 2 by Non-optimal, OCA, NMPC Algorithms (Rover 2)

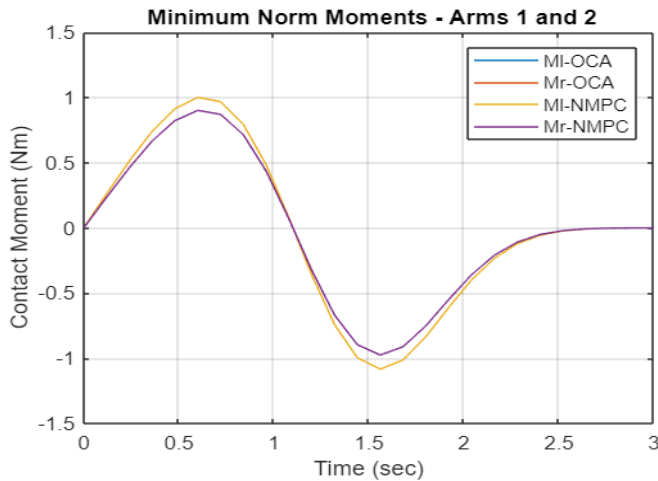


Figure 4: i Variation of the Contact Moments on Payload by Non-optimal, OCA, NMPC Algorithms (Arms 1 and 2)

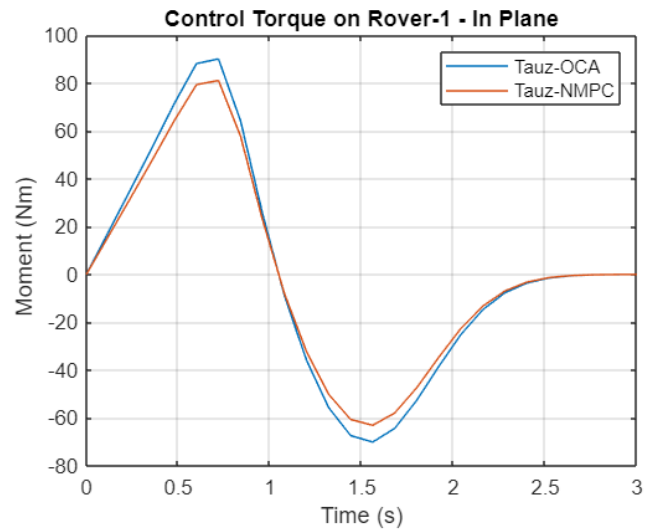


Figure 4: l Variation of the Control Moment on Rover 1 by Non-optimal, OCA, NMPC Algorithms (Rover 1)

Again, the NMPC is superior to OCA in obtaining minimum contact moments / forces applied to the common load while the two end-effectors are carrying a common load.

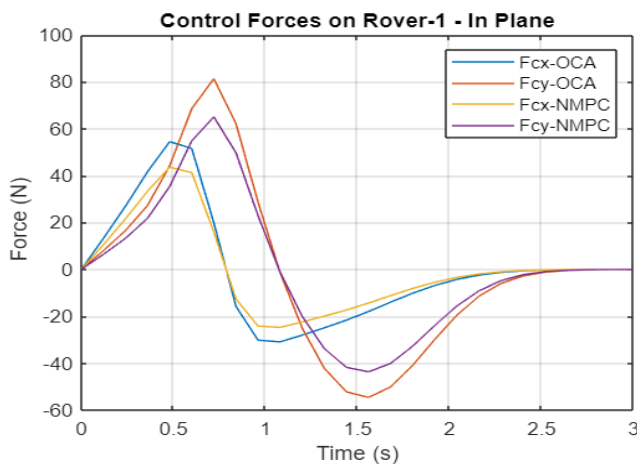


Figure 4: j Variation of the Control Forces on Rover 1 by Non-optimal, OCA, NMPC Algorithms (Rover 1)

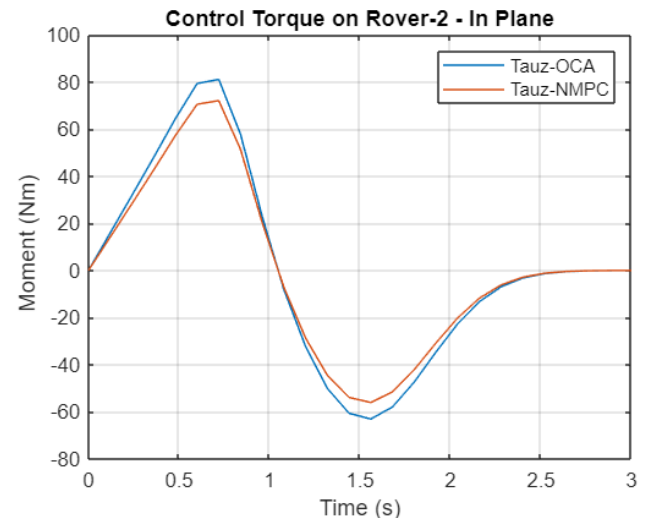


Figure 4: m Variation of the Control Moment on Rover 2 by Non-optimal, OCA, NMPC Algorithms (Rover 2)

A comparative analysis shows that again NMCP is superior to OCA in obtaining minimum norm of control moments and forces for Rovers 1 and 2.

The time variations of joint angular accelerations are shown in Figure 5.

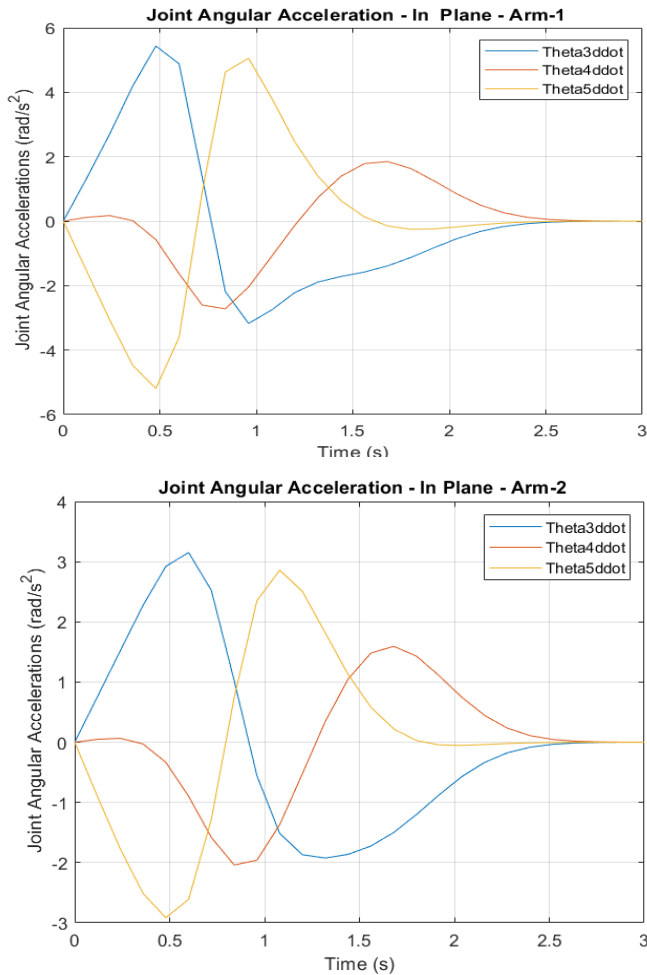


Figure 5: Variation of joint angular accelerations for the first and second arm

The joint accelerations are integrated to calculate rotational rates and angles using (13) and are presented in Figure 6.

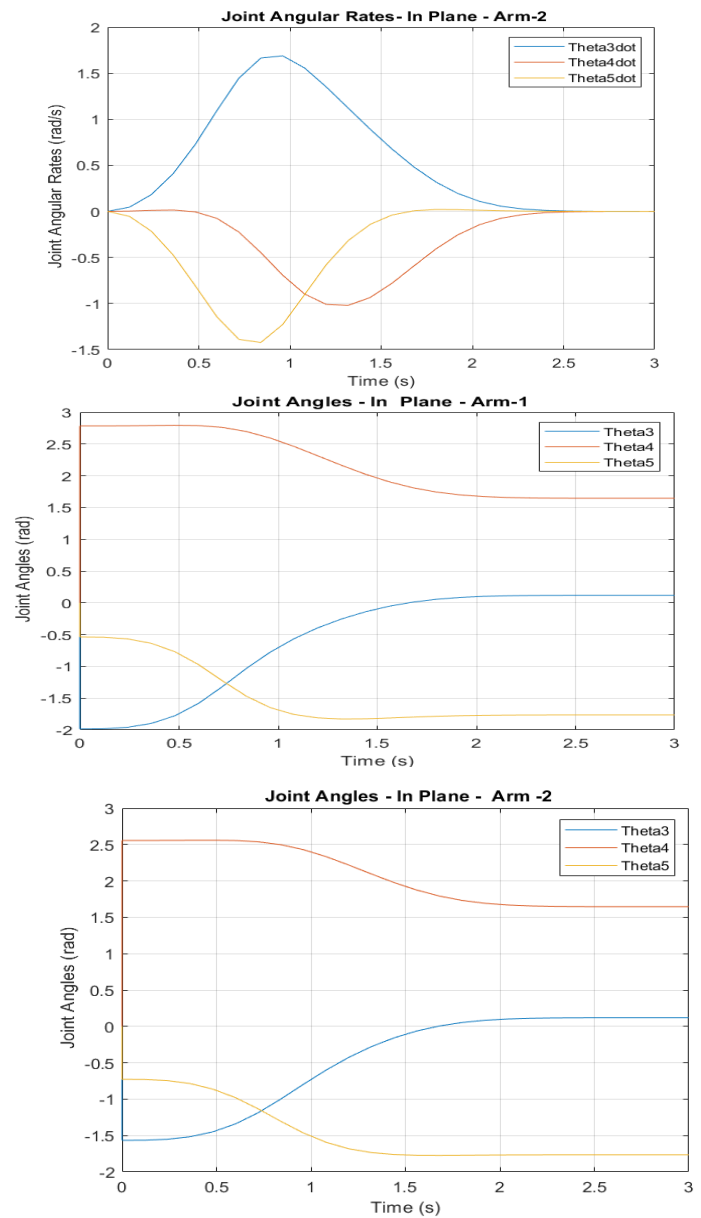
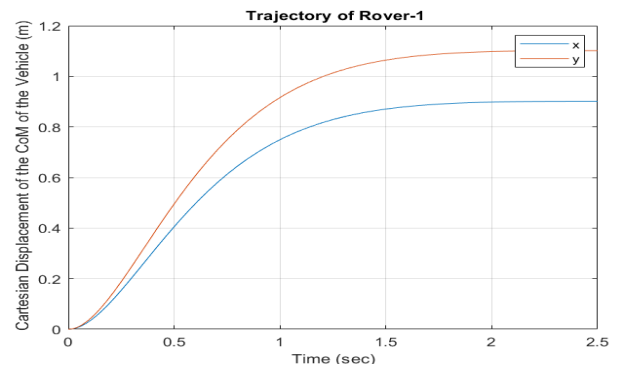
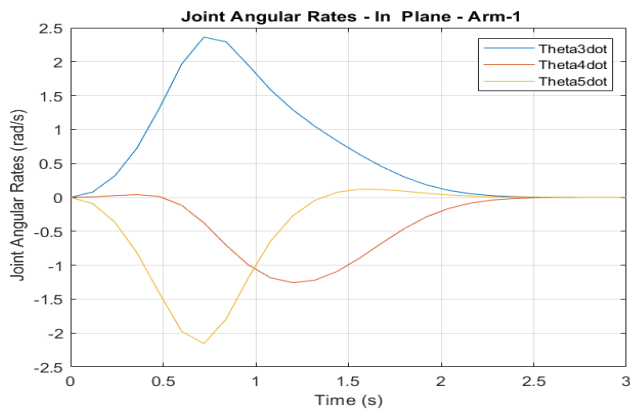


Figure 6: Variation of joint angular rates and angles for the first and second arm

The trajectories of the point C, the center of mass of the two rovers are determined by (22) and (8) and are shown in Figure 7.



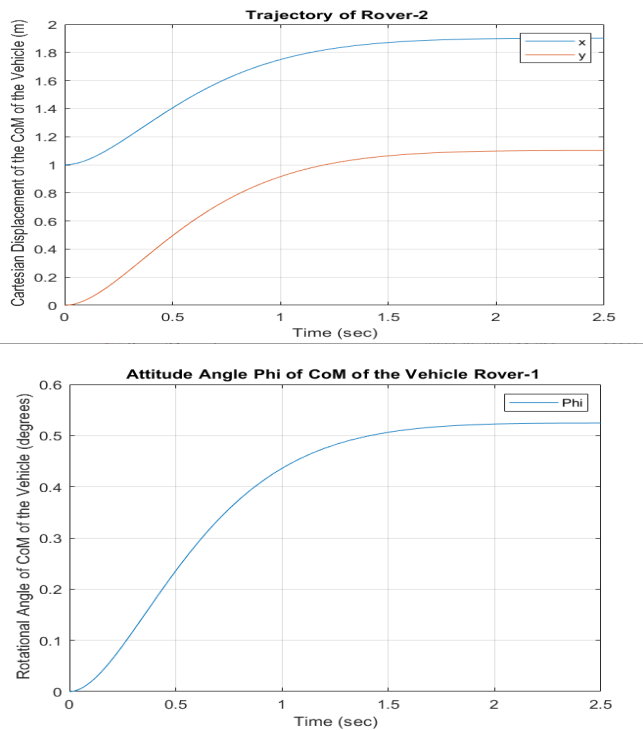


Figure 7: Variation of Rover 1 and 2 positions and orientations with time

The wheel angles of the two rovers are calculated utilizing (22) and are presented in Figure 8.

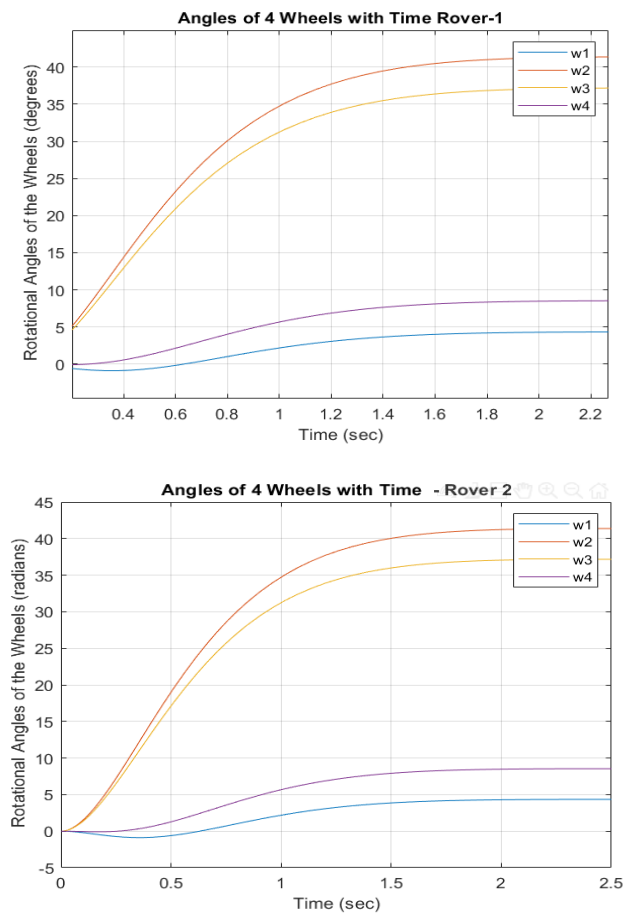


Figure 8 Variation of angles of rotations for rover wheels - rovers 1 and 2

## Conclusions and Future Work

The paper presented two novel control algorithms for motion and force control of a multi-rover robotics system when the two end-effectors carrying a common load. One algorithm is predicated on Optimal Control Allocation (OCA) and the other is a discretized (ii) Nonlinear Model Predictive Control (NMPC) algorithm.

The paper focused on developing robust and computationally efficient real-time control algorithms that can minimize the performance index consisting of the norm of the rovers control moments / forces, the joint torques, , as well as the contact moments / forces applied to the common load by two end-effectors.

The norm of wheel moments, joint torques, and the contact moments and forces were minimized to resolve the torque / moment saturation problem often seen while carrying a common load. The paper also presented a minimum norm solution for an underdetermined system subject to non-holonomic constraints. Moreover, the developed control algorithm also provided a real-time capability of trajectory for both the rovers and the arms while carrying a common load.

The system consisting of multi-rover with a dual arm was highly non-linear. The linear MPC technique on which most of the previous studies relied was not adequate. On the other hand, the computational complexity of a generic NMPC algorithm was very demanding. Therefore, in this paper, an elegant discretized technique with exact realization was implemented to take into account the full non-linear model and yet provide a simple real-time solution satisfying a minimum performance index subject to constraints.

The results of the computer simulations illustrated that the two algorithms OCA and NMPC worked efficiently. They were able to realize the minimum contact forces and moments and rover wheel moments and forces, joint torques, while manipulating a common load and tracking a reference load trajectory. In addition, the minimal norm solution also satisfied the non-holonomic constraints.

The results revealed that the optimization scheme used by the NMPC algorithm was the most effective when it came to achieving the lowest joint torques and forces. It was then followed by the OCA algorithm and the conventional least square method, respectively.

The authors are currently working on a research project to build a testbed to experimentally validate the computer simulation results. The comparisons of experimental and simulation results will be part of the future research work. Furthermore, the authors assumed no slippage occurred. However, the maximum driving force of each wheel is limited by the dynamic friction coefficient and the magnitude of the normal force acting on it. If this is exceeded, this assumption will no longer be valid. The normal forces will be incorporated in the dynamics model for the future work.

## Conflict of Interest

The authors declare no conflict of interest.

## References

- [1] S. Kalaycioglu, A. de Ruiter, "Coordinated Motion and Force Control of Multi-Rover Robotics System with Mecanum Wheels," in 2022 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), IEEE: 1–9, 2022, doi:10.1109/IEMTRONICS55184.2022.9795804.
- [2] D.S. Neculescu, B. Kim, S. Kalaycioglu, FREE MOTION, COLLISION AVOIDANCE AND CONTACT MOTION CONTROL FOR MOBILE ROBOTS, Elsevier: 223–228, 1993, doi:10.1016/B978-0-08-041897-1.50042-0.
- [3] N. Neculescu, B. Kim, S. Kalaycioglu, "Contact motion control for mobile robots," in 7th IFAC Symposium on Information Control Problems, IFAC, Toronto, 1992.
- [4] R. Fierro, F.L. Lewis, "Control of a nonholonomic mobile robot: backstepping kinematics into dynamics," in Proceedings of 1995 34th IEEE Conference on Decision and Control, IEEE: 3805–3810, doi:10.1109/CDC.1995.479190.
- [5] Yu Tian, N. Sidek, N. Sarkar, "Modeling and control of a nonholonomic Wheeled Mobile Robot with wheel slip dynamics," in 2009 IEEE Symposium on Computational Intelligence in Control and Automation, IEEE: 7–14, 2009, doi:10.1109/CICA.2009.4982776.
- [6] Y.H. Amengonu, Y.P. Kakad, "Dynamics and control for Constrained Multibody Systems modeled with Maggi's equation: Application to Differential Mobile Robots PartII," IOP Conference Series: Materials Science and Engineering, **65**, 012018, 2014, doi:10.1088/1757-899X/65/1/012018.
- [7] G. Campion, B. d'Andrea-Novell, G. Bastin, Controllability and state feedback stabilizability of non holonomic mechanical systems, Springer-Verlag, Berlin/Heidelberg: 106–124, doi:10.1007/BFb0039268.
- [8] A.M. Bloch, N.H. McClamroch, "Control of mechanical systems with classical nonholonomic constraints," in Proceedings of the 28th IEEE Conference on Decision and Control, IEEE: 201–205, doi:10.1109/CDC.1989.70103.
- [9] S. Kalaycioglu, "Control of multiple robot manipulators with optimal force distribution," in IEEE Canadian Conference on Electrical and Computer Engineering, 1991.
- [10] M. Vukob, S. Gros, G. Horn, G. Frison, K. Geebelen, J.B. Jørgensen, J. Swevers, M. Diehl, "Real-time nonlinear MPC and MHE for a large-scale mechatronic application," Control Engineering Practice, **45**, 64–78, 2015, doi:10.1016/j.conengprac.2015.08.012.
- [11] J.B. Rawlings, "Tutorial overview of model predictive control," IEEE Control Systems, **20**(3), 38–52, 2000, doi:10.1109/37.845037.
- [12] Y. Shi, K. Zhang, "Advanced model predictive control framework for autonomous intelligent mechatronic systems: A tutorial overview and perspectives," Annual Reviews in Control, **52**, 170–196, 2021, doi:10.1016/j.arcontrol.2021.10.008.
- [13] P.D. Christofides, R. Scattolini, D. Muñoz de la Peña, J. Liu, "Distributed model predictive control: A tutorial review and future research directions," Computers & Chemical Engineering, **51**, 21–41, 2013, doi:10.1016/j.compchemeng.2012.05.011.
- [14] M. Ellis, H. Durand, P.D. Christofides, "A tutorial review of economic model predictive control methods," Journal of Process Control, **24**(8), 1156–1178, 2014, doi:10.1016/j.procont.2014.03.010.
- [15] F. Michael, Implementation of Linear Model Predictive Control –Tutorial, 2021.
- [16] S. Yu, M. Reble, H. Chen, F. Allgöwer, "Inherent Robustness Properties of Quasi-infinite Horizon MPC," IFAC Proceedings Volumes, **44**(1), 179–184, 2011, doi:10.3182/20110828-6-IT-1002.01969.
- [17] H. Wei, C. Shen, Y. Shi, "Distributed Lyapunov-Based Model Predictive Formation Tracking Control for Autonomous Underwater Vehicles Subject to Disturbances," IEEE Transactions on Systems, Man, and Cybernetics: Systems, **51**(8), 5198–5208, 2021, doi:10.1109/TSMC.2019.2946127.
- [18] H. Wei, Q. Sun, J. Chen, Y. Shi, "Robust distributed model predictive platooning control for heterogeneous autonomous surface vehicles," Control Engineering Practice, **107**, 104655, 2021, doi:10.1016/j.conengprac.2020.104655.
- [19] K. Zhang, Q. Sun, Y. Shi, "Trajectory Tracking Control of Autonomous Ground Vehicles Using Adaptive Learning MPC," IEEE Transactions on Neural Networks and Learning Systems, **32**(12), 5554–5564, 2021, doi:10.1109/TNNLS.2020.3048305.
- [20] Y. Zou, X. Su, S. Li, Y. Niu, D. Li, "Event-triggered distributed predictive control for asynchronous coordination of multi-agent systems," Automatica, **99**, 92–98, 2019, doi:10.1016/j.automatica.2018.10.019.
- [21] K. Zhang, Y. Shi, "Adaptive model predictive control for a class of constrained linear systems with parametric uncertainties," Automatica, **117**, 108974, 2020, doi:10.1016/j.automatica.2020.108974.
- [22] J.S. Ladoiye, D.S. Neculescu, J. Sasiadek, "Force Control of Surgical Robot with Time Delay using Model Predictive Control," in Proceedings of the 15th International Conference on Informatics in Control, Automation and Robotics, SCITEPRESS - Science and Technology Publications: 202–210, 2018, doi:10.5220/0006908602020210.
- [23] R.A. Gangapersaud, G. Liu, A.H.J. de Ruiter, "Detumbling of a non-cooperative target with unknown inertial parameters using a space robot," Advances in Space Research, **63**(12), 3900–3915, 2019, doi:10.1016/j.asr.2019.03.002.
- [24] T. Englert, A. Völz, F. Mesmer, S. Rhein, K. Graichen, "A software framework for embedded nonlinear model predictive control using a gradient-based augmented Lagrangian approach (GRAMPC)," Optimization and Engineering, **20**(3), 769–809, 2019, doi:10.1007/s11081-018-9417-2.
- [25] K. Rathai, Synthesis and Real-time Implementation of Parameterized NMPC Schemes for Automotive Semi-active Suspension Systems, PhD Thesis, Communauté Universit'e Grenoble Alpes, Grenoble, 2020.
- [26] R. Quirynen, M. Vukob, M. Zanon, M. Diehl, "Autogenerating microsecond solvers for nonlinear MPC: A tutorial using ACADO integrators," Optimal Control Applications and Methods, **36**(5), 685–704, 2015, doi:10.1002/oca.2152.
- [27] F. Aghili, "Optimal control of a space manipulator for detumbling of a target satellite," in IEEE Int. Conf. Robot. Automatica, IEEE, 2009.
- [28] T. Rybus, J. Seweryn, J. Sasiadek, "Application of predictive control for manipulator mounted on a satellite," Archives of Control Sciences, **28**(1), 105–118, 2018.
- [29] M. Wang, J. Luo, U. Walter, "A non-linear model predictive controller with obstacle avoidance for a space robot," Advances in Space Research, **57**(8), 1737–1746, 2016, doi:10.1016/j.asr.2015.06.012.
- [30] M. Morato, J. Normey-Rico, O. Sename, "Model Predictive Control Design for Linear Parameter Varying Systems: A Survey," in Annual Reviews in Control, 64–80, 2020.
- [31] E. Psomiadis, E. Papadopoulos, "Model-Based/Model Predictive Control Design for Free Floating Space Manipulator Systems," in 2022 30th Mediterranean Conference on Control and Automation (MED), IEEE: 847–852, 2022, doi:10.1109/MED54222.2022.9837196.
- [32] M. Wada, S. Mori, "Holonomic and omnidirectional vehicle with conventional tires," in Proceedings of IEEE International Conference on Robotics and Automation, IEEE: 3671–3676, doi:10.1109/ROBOT.1996.509272.
- [33] J. Ostrowski, J. Burdick, "The Geometric Mechanics of Undulatory Robotic Locomotion," The International Journal of Robotics Research, **17**(7), 683–701, 1998, doi:10.1177/027836499801700701.
- [34] C. Stöger, A. Müller, H. Gattringer, Parameter Identification and Model-Based Control of Redundantly Actuated, Non-holonomic, Omnidirectional Vehicles, 207–229, 2018, doi:10.1007/978-3-319-55011-4\_11.
- [35] P.F. Muir, C.P. Neuman, "Kinematic modeling of wheeled mobile robots," Journal of Robotic Systems, **4**(2), 281–340, 1987, doi:10.1002/rob.4620040209.
- [36] F.G. Pin, S.M. Killough, "A new family of omnidirectional and holonomic wheeled platforms for mobile robots," IEEE Transactions on Robotics and Automation, **10**(4), 480–489, 1994, doi:10.1109/70.313098.
- [37] G. Campion, G. Bastin, B. D'Andrea-Novell, "Structural properties and classification of kinematic and dynamic models of wheeled mobile robots," in [1993] Proceedings IEEE International Conference on Robotics and Automation, IEEE Comput. Soc. Press: 462–469, doi:10.1109/ROBOT.1993.292023.
- [38] G. Wampfler, M. Salecker, J. Wittenburg, "Kinematics, Dynamics, and Control of Omnidirectional Vehicles with Mecanum Wheels," Mechanics of Structures and Machines, **17**(2), 165–177, 1989, doi:10.1080/15397738909412814.
- [39] A. Gfrerrer, "Geometry and kinematics of the Mecanum wheel," Computer Aided Geometric Design, **25**(9), 784–791, 2008, doi:10.1016/j.cagd.2008.07.008.
- [40] L.-C. Lin, H.-Y. Shih, "Modeling and Adaptive Control of an Omni-Mecanum-Wheeled Robot," Intelligent Control and Automation, **04**(02), 166–179, 2013, doi:10.4236/ica.2013.42021.

- [41] A. Shimada, S. Yajima, P. Viboonchaicheep, K. Samura, "Mecanum-wheel vehicle systems based on position corrective control," in 31st Annual Conference of IEEE Industrial Electronics Society, 2005. IECON 2005., IEEE: 6 pp., 2005, doi:10.1109/IECON.2005.1569224.
- [42] Y. Wang, D. Chang, "Motion performance analysis and layout selection for motion system with four Mecanum wheels," *Journal of Mechanical Engineering*, **45**(5), 307–316, 2009.
- [43] M.O. Tatar, C. Popovici, D. Mandru, I. Ardelean, A. Plesa, "Design and development of an autonomous omni-directional mobile robot with Mecanum wheels," in 2014 IEEE International Conference on Automation, Quality and Testing, Robotics, IEEE: 1–6, 2014, doi:10.1109/AQTR.2014.6857869.