Available online at www.sciencedirect.com
ScienceDirect
Procedia CIRP 40 (2016) 407 – 412
www.elsevier.com/locate/procedia

13th Global Conference on Sustainable Manufacturing - Decoupling Growth from Resource Use

Using Adaptive Model Predictive Technique to Control Underactuated Robot and Minimize Energy Consumption

Ahmad Albalasie[a,*], Guenther Seliger[a], Ahmed Abu Hanieh[b]

[a]*Technical University of Berlin, Pascalstr. 8-9, Berlin 10587, Germany*
[b]*Birzeit University, Birzeit, Palestine*

* Corresponding author. Tel.: +49 (0)30 / 314-21255; fax: +49 (0)30 / 314-22759. *E-mail address:* albalasie@mf.tu-berlin.de

Abstract

This paper presents an adaptive model predictive control scheme to control the underactuated and redundant robot, the robot has highly nonlinear coupling because of the existence of a passive axis. Adaptive model predictive control provides a framework to solve optimal discrete control problem for a nonlinear system under input saturation and state constraints. The optimal reference trajectory is computed by using Quasi-linearization (QL) approach to minimize the energy consumption for underactuated motion between two points. The challenge is to meet the performance requirements e.g. position accuracy, repeatability, and precision, combined with high speed capability. Numerical simulations are conducted to validate the control scheme. Simulation results show very good comparison and prove the adequateness of this control technique for underactuated industrial robots.

Peer-review under responsibility of the International Scientific Committee of the 13th Global Conference on Sustainable Manufacturing

Keywords: adaptive model predictive control; underactuated motion; industrial robot; trajectory tracki

1. Introduction

Industrial robots are key factors in implementing the production on the desired scale, speed, quality, and cost. The President of the International Federation of Robotics (IFR) announced in September - 2014 “more than 200,000 industrial robots will be installed in 2014 worldwide, 15% more than in 2013” [1]. Furthermore, he stated “the accelerating demand for industrial robots will continue between 2015 and 2017, growth will likely continue at about 12% on average per year” [1]. This statement emphasizes the need for continuous and intensive research and development in the various fields associated with industrial robots.

Energy efficiency is a field of research in manufacturing and robotics engineering. Since the prices of resources in general and crude oil or gas in specific are increasing, the research focus on saving energy within your production processes became increasingly interesting for companies manufacturing or assembling goods. This emphasizes the great potential to reduce the energy consumption of industrial robots. Consequently minimizing the energy consumption (EC) for the robot is an important issue because also it is minimizing the $\mathbf{CO_2}$ emission in the production stage of a product’s life cycle, decreases the cost of the products and it increases the contribution value of the production.

Several methods have been developed and tested for implementing energy efficient processes in the last decade. Some of these methods are managerial and the others are technical. The managerial methods include: overall production optimization strategies, and strategic and intelligent use of the robot application. The technical methods consist of intelligent braking management systems, the temporal storage of energy in a capacitive buffers, and the optimizing of robot trajectories [2–6].

SAMARA is a prototype of industrial robot for material handling process. It has five degrees of freedom as shown in Fig.1. Furthermore, it uses redundant, underactuated configurations and constrained optimization algorithm to minimize the energy consumption during executing the handling operations [2,7,8,3,3].

Peer-review under responsibility of the International Scientific Committee of the 13th Global Conference on Sustainable Manufacturing
doi:10.1016/j.procir.2016.01.080

Through the previous phase of the work, the second generation of SAMARA prototype has been designed and built at TU Berlin. This new type of robot was designed and path planned minimize the energy consumption with high speed capability and high payload ratio. SAMARA prototype was programmed using optimal path planning algorithm based on the evolutionary algorithm [7,8,3] or the quasi-linearization algorithm in order to minimize the energy consumption during a specific cycle time [2].

In particular, SAMARA has two phases of motion to execute the required tasks. The first phase is the null space phase where the end effector executes the pick and place tasks. Consequently, all the motors are active in this phase. The angular position, angular velocity, and the angular acceleration have been computed for the null space motion by using the analytical solution for each axis. The second phase is the underactuated motion where the end effector moves from point A in the station to point B in the conveyor belt or the vice versa as shown in Fig.2. In this case, the third axis is a passive i.e. un-motorized axis. Also, the boundary values for the underactuated motion are known. This problem is known in mathematics, and it is called two-point boundary value problem (TPBVP).

The handling robots have the property of a relative long movement distances between the processing points of the handling operation. Although these robots consumes high energy consumption in movements among the other robot types, the energy saving in such robots is still larger than the other ones [5].

This paper uses new kinematics for underactuated motion (UAM) [7] and new type of control to execute the pick and place tasks uses a novel method for solving a tracking problem by using the adaptive model predictive controller based on trajectory tracking (AMPCTT) to control underactuated industrial robot taking into account increasing the energy saving by using the (QL) as approach for trajectory planning. This paper is an extension work for a previous work which was developed by TU Berlin and Birzeit university researchers for trajectory planning of UAM [3].

Robots commonly have fast and nonlinear dynamics, the implementation of MPC remains fundamentally limited due to high demand in computational resource associated with optimization. Though most physical systems are inherently nonlinear in nature, the majority of MPC applications are based on the linear dynamic model, mainly to take the computational advantages of MPC.

There are two main types of MPCs. The first type is the linear MPC. It uses the linear model for describing the nonlinear dynamical model for the system at the specific operating point. On the contrary, the second class is the nonlinear MPC (NMPC) which uses a nonlinear model. The computational time for the linear MPC is less than NMPC because of its linearity. But if the nonlinearity of the system is too high, the linear MPC cannot work accurately. Otherwise, NMPC calculates more accurate results, but it consumed longer time in calculations which is not suitable in some applications in the mode of real time environment. Researchers studied the control by using MPC or NMPC based on tracking reference paths for different applications [9–13]. The problem of nonlinearity in the dynamic equations of the underactuated robots causes a problem in the performance, stability, and the other requirements such as: the accuracy, repeatability, and the precision. According to [14], "MPC predicts future behavior using a linear-time-invariant (LTI) dynamic model. In practice, such predictions are never exact, and a key tuning objective is to make MPC insensitive to prediction errors. In many applications, this is sufficient for robust controller performance. If the plant is strongly nonlinear or its characteristics vary dramatically with time, LTI prediction accuracy might degrade so much that MPC performance becomes unacceptable".

This paper suggests another solution for the previous problem by using the adaptive MPC (AMPC). AMPC has the ability to update the linearized model at each control interval, therefore a set of LTI approximation at each current operating condition is used to approximate the nonlinear model and this will improve the response of the robot [14].

Fig. 1 second generation of SAMARA robot

There are two concepts applied in this research to minimize the energy consumption:

- The robot uses new type of kinematics is the underactuated and redundant configurations. The benefit of this type of configuration is that the number of joints are more than the number of actuators, and it uses the principle of momentum conservation [7].
- Minimize the energy consumption by using any optimization algorithm for trajectory planning. To implement this idea the researchers at TU Berlin

developed two approaches of trajectory planning for SAMARA prototype, one of them is the evolutionary algorithm (EA) to solve the TPBVP in the phase of UAM [8,3] and the second approach by uses QL algorithm to solve the TPBVP [2]. However there are some advantages and disadvantages for each one of them for example minimizing the EC by using EA is better than minimizing EC by using QL, on the other hand, the convergence rate for QL is faster than EA so it reduces the computation time dramatically.

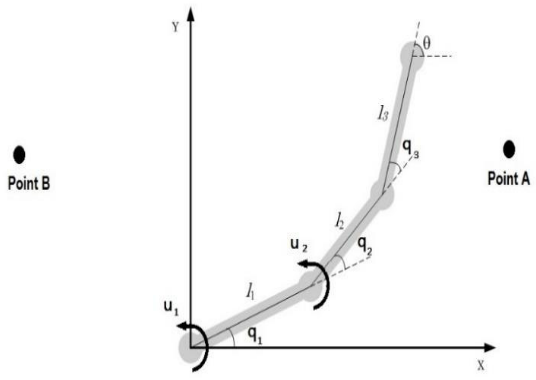

Fig. 2 SAMARA configuration with Point A and Point B in the phase of UAM.

The organization of this paper is as follows. State space representation for the dynamical equations of the robot in phase of UAM is shown in section 2. Control scheme for UAM by using AMPC is shown in section 3. Section 4 contains two numerical examples to control SAMARA prototype to validate the work. Finally, conclusions and future works are shown in section 5.

2. State Space Representation for UAM

The dynamic equations for the horizontal planer underactuated manipulator are shown in (1).

$$M(q)\ddot{q} + C(q,\dot{q}) + D(\dot{q}) = U \tag{1}$$

where:

- v: is the number of axes in the robot.
- $M(q) \in R^{v*v}$: the inertia matrix.
- $C(q,\dot{q}) \in R^{v}$: Coriolis and centrifugal forces.
- $D(q,\dot{q}) \in R^{v}$: the damping and friction moments.
- $U \in R^{v}$: the motor torques.
- $q \in R^{v}$: the generalized coordinates for all axes.

As a consequence (2) can be expressed in more details about the active and passive axes for SAMARA prototype as shown below:

$$\begin{bmatrix} m_{aa}(q) & m_{ap}(q) \\ m_{pa}(q) & m_{pp}(q) \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} c_1(q,\dot{q}) \\ c_2(q,\dot{q}) \end{bmatrix} + \begin{bmatrix} d_1(\dot{q}) \\ d_2(\dot{q}) \end{bmatrix} = \begin{bmatrix} T \\ 0 \end{bmatrix} \tag{2}$$

where:

- a: number of active axes.
- p: number of passive axes.
- $q_1 \in R^{a}, q_2 \in R^{p}$: the generalized coordinates for the active and passive axes respectively.
- $c_1(q,\dot{q}) \in R^{a}$, $c_2(q,\dot{q}) \in R^{p}$: Coriolis and centrifugal forces for active and passive axes respectively.
- $d_1(\dot{q}) \in R^{a}, d_2(\dot{q}) \in R^{p}$: damping and friction moments for the active and passive axes respectively.
- $T \in R^{a}$: vector containing motor torques.

The state space representation for the system in phase of UAM is shown in (3).

$$x_1 = q \quad , \quad x_2 = \dot{x}_1 = \dot{q} \quad , \quad \dot{x}_2 = \ddot{q}$$
$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ M^{-1}(x_1) * (U - C(x_1,x_2) - D(x_2)) \end{bmatrix} = f(x,u) \tag{3}$$

where: $U = \begin{bmatrix} T \\ 0 \end{bmatrix}$

In conclusion the model is highly nonlinear model, which is difficult to be controlled but reference [15] proves that it is controllable.

3. Control Scheme

The control scheme for SAMARA prototypes contains three main components: AMPC, optimizer and successive linearization block, as shown in Fig.3. The following subsections describe the design and the task for each component.

3.1 The Optimizer for the Path Planning

There are two algorithms for trajectory planning developed at TU Berlin, EA and QL as mentioned in the previous section. In this research, QL is used as a trajectory planning because it is faster than EA. Trajectory planning for minimizing the EC for UAM will be called optimizer, and these simulations are usually done in offline mode. Minimizing the EC during the UAM has been executed by minimizing the input torques for each joint as shown in (4).

$$J = \min \frac{1}{2}\int_{t_0}^{t_f} U^T * R * U \tag{4}$$
$$\text{sub to } \dot{x} = f(x(t), u(t), t)$$

Minimizing the cost function with respect to state space constraints can be calculated by minimizing the Hamiltonian equation as shown in (5) [16].

$$\mathcal{H} = U^T * R * U + \sum_{i=1}^{n} \lambda_i * f(x(t), u(t), t)_{\,i} \tag{5}$$

where:-

- n is the number of states.
- m is the number of the inputs.
- R is a positive definite matrix
- $\lambda \in R^{n}$ is the Lagrange multipliers functions, also called the co-state variables.

By following the steps which are mentioned in a previous work for the authors in [2] the optimal trajectories is computed easily. These trajectories are the desired references r(t) for the joints of the robot see Fig.4 and Fig.6. This algorithm calculates the angular position, the angular velocity, the angular acceleration, and the expected torque for each axis to minimize the EC.

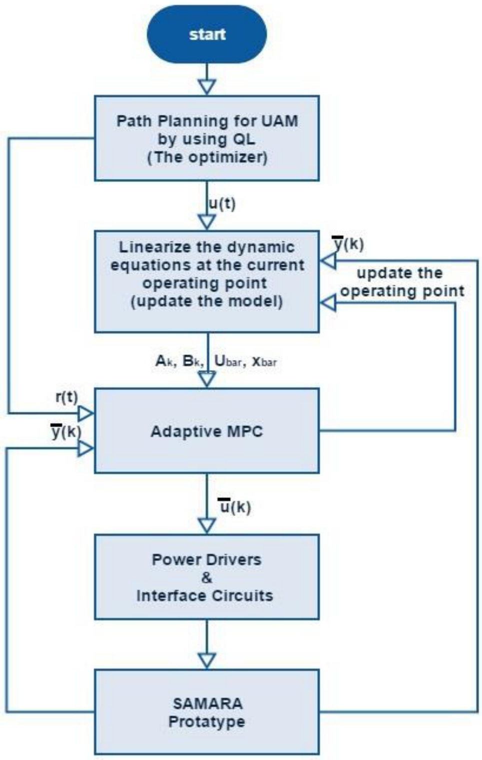

Fig.3 Control flowchart

3.2 Successive Linearization Process

The state space representation of the robot is a nonlinear model as shown in (3) which is unsuitable for AMPC because it uses the LTI models, in order to solve this successive linearization is used. Consequently, the main task of successive linearization is to provide an LTI approximation at the current operating point at each control interval for AMPC to approximate the nonlinearity.

Nominal optimal references for the angular position, angular velocity, angular acceleration, and torque for each joint have been computed by using the optimizer as shown in the previous section. However, AMPC uses LTI models for the system to track the nominal optimal references, therefore the system linearized around initial operating point then it is linearized around nominal trajectories u_k, x_k. To sum up, this algorithm starts by applying the nominal control sequence u_k. The response is a nominal trajectory y_k as an open loop response. Even so, to improve the response the trajectories are linearized around u_k, x_k and the modified problem is solved. The plant model can be written in terms of the error as shown in (6) and (7). Additionally, the state space models discretized because AMPC needs discrete models.

$$\bar{x}_{k+1} = A_k\bar{x}_k + B_k\bar{u}_k \quad (6)$$
$$\bar{y}_{k+1} = C_k\bar{x}_k + D_k\bar{u}_k \quad (7)$$

where:

- $A_k = \begin{bmatrix} \frac{\partial f(x,u)}{\partial x_1} & \frac{\partial f(x,u)}{\partial x_2} & \dots & \frac{\partial f(x,u)}{\partial x_n} \end{bmatrix}_{n*n}$ (8)
- $B_k = \begin{bmatrix} \frac{\partial f(x,u)}{\partial u_1} & \frac{\partial f(x,u)}{\partial u_2} & \dots & \frac{\partial f(x,u)}{\partial u_m} \end{bmatrix}^T_{n*m}$ (9)
- C_k is identity matrix and D_k is a zero matrix.
- $\bar{x}_k$ represents the error with respect to the reference points
- $\bar{u}_k$ is associated to control input.

3.3 Adaptive Model Predictive Control

AMPC, MPC, and NMPC have the abilities to predict the future response of the plant in a specified horizon not like PID controller or LQR controller. Nevertheless, they use only the current values for the input. The most exciting properties from control point of view such as robustness properties in inputs and on the states are satisfied in this type of controllers [17]. In addition, MPC family can deal naturally with constraints on the input, and on the states [14,12].

MPC solves the quadratic optimization (QP) optimization problem to compute the input control values, usually the models for the dynamic equations are nonlinear models but the model linearized at specific operating point. In some cases, the influence of nonlinear dynamics effects is so important that the use of nonlinear model predictive control (NMPC) is unavoidable [18]. However, AMPC has the solution for the previous problem because it has an interesting feature which is updating the model at each control interval. As a result of that, LTI model used at each current operating point to approximate the nonlinear model accurately. AMPC calculates the optimal control sequence by solving QP problem to find the minimal value for a specific cost function ($\emptyset$) shown in (10), and the constraints are shown in (11-13).

$$\emptyset = \sum_{j=1}^{P}\left[\left(r_{k+j} - y_{k+j}\right)^T Q\left(r_{k+j} - y_{k+j}\right) + \bar{u}_{k+j}^T R\bar{u}_{k+j} + \Delta\bar{u}_{k+j}^T S\Delta\bar{u}_{k+j}\right] \quad (10)$$
$$\bar{x}_{k+1} = A_k\bar{x}_k + B_k\bar{u}_k \quad (11)$$
$$\Delta\bar{u}_{min} < \Delta\bar{u}_{k+j-1} < \Delta\bar{u}_{max} \quad (12)$$
$$\bar{u}_{min} < \bar{u}_{k+j-1} < \bar{u}_{max} \quad (13)$$

The notation $\bar{x}_{k+j}$ indicates the value of $\bar{x}$ at the instant $k + j$ predicted at instant k.

where:

- P is the prediction horizon.
- C is the control horizon.
- Q, R, S are weighting matrices, with $Q \geq 0$ and $R, S > 0$.

AMPC uses the direct approach by finding the optimal control in such a way that constraints are not violated. Additionally, the control scheme designed by using Model Predictive Control Toolbox and Simulink Toolbox provided by MATLAB software.

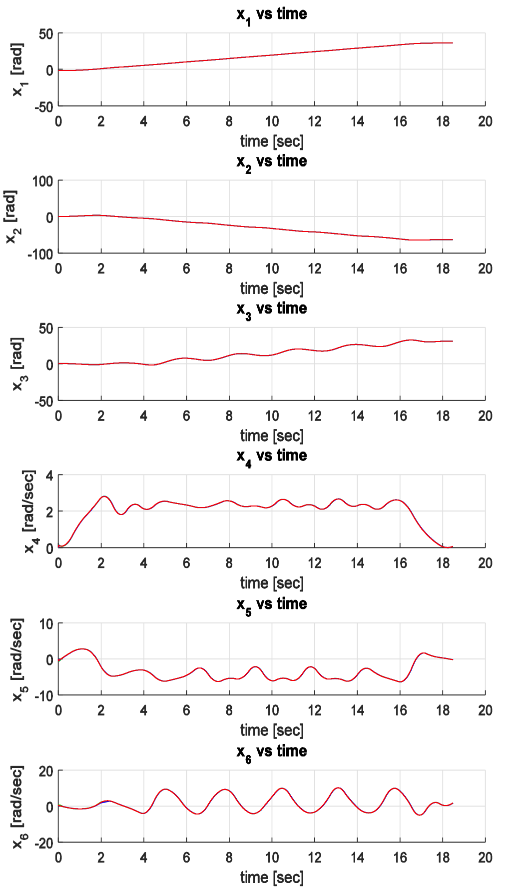

Fig. (4) reference trajectory (red), AMPC response (green), and state estimation (blue)

4. Simulation and Numerical Results

The proposed AMPC scheme has been tested on tracking reference trajectories to validate the control scheme. The reference trajectory contains many phases such as start motion, the null space motion which is computed by the analytical solution to execute the pick and place tasks, the UAM which is computed by QL to move from point A to point B or the vice versa, and the end motion. Two numerical examples clarify the response of AMPC with comparison to the reference trajectories and the state estimator.

Case 1:

Reference trajectories with five cycles for pick and place has been computed by using QL for UAM and the analytical solution for the null space motion. These trajectories are the desired response for the robot. As a result of that, the response of the robot should be track these trajectories.

The simulation response of the robot by using the AMPC has been calculated to test the ability of the controller can handle the nonlinear kinematics as well as the multivariable coupling in the system dynamics by tracking the reference trajectories. The prediction horizon for AMPC is P=10 and the control horizon C=1 whereas the sampling time $T_s = 2$ ms.

As shown in Fig. (4) AMPC tracks the reference trajectories accurately. In addition, the total power consumed for the reference trajectories and the total power consumed for AMPC is shown in Fig. (5).

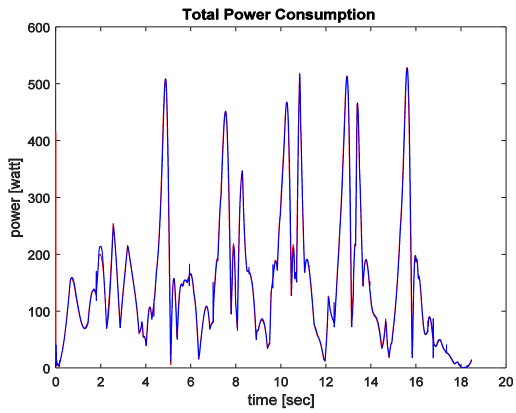

Fig. (5) total power consumption for the reference trajectories (red), and total power consumption for AMPC (blow)

Case 2:

Reference trajectories with one cycle for pick and place has been computed. The reference trajectories, the simulation response of AMPC, and the state estimator are shown in Fig. (6). The prediction horizon for the AMPC is P=10 and the control horizon is C=5 whereas the sampling time $T_s = 2$ ms:

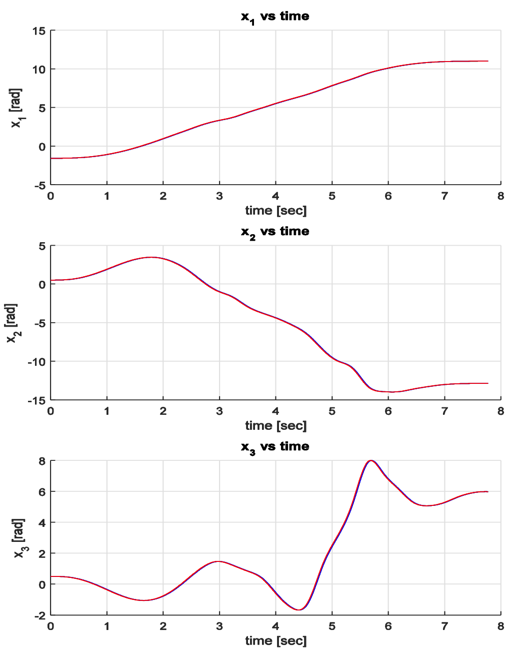

Fig. (6) reference trajectory (red), AMPC response (green), and state estimation (blue)

5. Conclusion and future work

This paper presented control scheme by using AMPC based on QL trajectory tracking to minimize the EC. This type of control can deal with nonlinearities in the model because the controller updates the model at each control interval. Moreover, it has the ability to predict the future response such that it solves the expected problems. The future works will focus on testing the robustness of the controller and the effect of the uncertainty in some model parameters on the presented control scheme. Finally, QL approach developed to minimize the energy consumption and the cycle time to improve the productivity and the energy efficiency.

As shown in Fig. (4) and in Fig. (6) AMPC tracks the reference trajectories accurately.

References

[1] International Federation of Robotics, 2014. World Robotics. http://www.worldrobotics.org/index.php?id=home&news_id=274.

[2] Albalasie A., Glodde A., Seliger G., Abu Hanieh A. (Eds.), 2014. Quasi-linearization Approach for the Under-actuated Robots. Elsevier.

[3] Brett T., Quiel H., Seliger G. (Eds.), 2010. Eco-efficient path planning for a redundant underactuated assembly robot, 5 pp.

[4] Dāvis M., 2013. Increasing Energy Efficiency of Robotized Production Systems in Automobile Manufacturing: Summary of doctoral thesis, 32 pp.

[5] Dāvis M., Leonids R. (Eds.), 2011. Recuperated Energy Savings Potential and Approaches in Industrial Robotics. IEEE, Piscataway, N.J., 1 online resource (835).

[6] Paes K., Dewulf W., Elst K.V., Kellens K., Slaets P., 2014. Energy Efficient Trajectories for an Industrial ABB Robot. Procedia CIRP 15, 105–110.

[7] Brett T., Seliger G. (Eds.), 2008. New kinematics and motion approach for parts handling, 6 pp.

[8] Brett T., 2013. Effective motion design applied to energy efficient handling. Fraunhofer Verlag.

[9] Castillos C., Moreno W., Valavani K. (Eds.), 2007. Unmanned Helicopter Waypoint Trajectory Tracking Using Model Predictive Control. IEEE, [Piscataway, N.J.], 1 online resource.

[10] Connette C.P., Hofmeister S., Bubeck A., Haegele M., Verl A. (Eds.), 2010. Model-Predictive Undercarriage Control for a Pseudo-Omnidirectional, Wheeled Mobile Robot: Date, 7-9 June 2010. [VDE Verlag], [Berlin], 1 online resource.

[11] Fahimi F., 2007. Non-linear model predictive formation control for groups of autonomous surface vessels. International Journal of Control 80 (8), 1248–1259.

[12] Mayne D.Q., Rawlings J.B., Rao C.V., Scokaert P., 2000. Constrained model predictive control: Stability and optimality. Automatica 36 (6), 789–814.

[13] Oh S.-R., Sun J., 2010. Path following of underactuated marine surface vessels using line-of-sight based model predictive control. Ocean Engineering 37 (2-3), 289–295.

[14] Math Works, 2014. Adaptive MPC. Math Works. http://de.mathworks.com/help/mpc/ug/adaptive-mpc.html. Accessed 24 February 2015.

[15] Arai H. (Ed.), 1996. Controllability of a 3-DOF manipulator with a passive joint under a non-holonomic constraint. IEEE, Piscataway, 1 online resource (4004.

[16] Kirk D.E., 2004. Optimal control theory: An introduction. Dover Publications, Mineola, N.Y., 1 online resource (ix, 452.

[17] Magni L., Sepulchre R., 1997. Stability margins of nonlinear receding-horizon control via inverse optimality. Systems & Control Letters 32 (4), 241–245.

[18] Tricaud C., Yang Q.C. (Eds.), 2008. Linear and nonlinear model predictive control using a general purpose optimal control problem solver RIOTS 95. IEEE, Piscataway, NJ, Online-Resource.