Experimental Comparison of Adaptive Controllers for Trajectory Tracking in Agricultural Robotics

Jarle Dørum, Tryve Utstumo, Jan Tommy Gravdahl
Department of Engineering Cybernetics
Norwegian University of Science and Technology
NO-7491 Trondheim, Norway

***Abstract*—This paper describes the development of several controllers to handle a trajectory tracking problem for a differentially wheeled robot. Both simulations and tests on a real robot were performed. A simple kinematic controller has been implemented to calculate desired velocities based on current position and trajectory. In order to also consider the current velocities, i.e. the dynamics of the system, the output of this controller was used as input to a dynamic controller derived from a nonlinear model. The dynamic controller was made adaptive by using an on-line parameter estimation scheme to estimate the unknown parameters of the nonlinear model. Lastly, a direct model reference adaptive controller (MRAC) based on a linear model was derived and implemented as an alternative to the adaptive dynamic controller.**

***Index Terms*—Mobile robots, differentially wheeled robots, trajectory tracking, adaptive control, system identification, agricultural robotics,**

I. Introduction

This paper presents part of the ongoing research for developing an agricultural robot that autonomously navigates in row crops while identifying and precision spraying individual weed leaves with herbicide. The robot is a differentially steered robot with two rear mounted caster wheels, and may be modeled as a unicycle-like robot. A picture of the prototype during testing in row crops is shown in Fig. 1.

Previous research on the project includes development of a precision drop-on-demand nozzle for herbicide application [1], a model predictive row controller [2] to minimize potential crop damage during operation and attitude estimation in agricultural robotics [3].

The nozzle array presented in [1] is intended to only be slightly wider than the row crops, meaning that the robot has to follow the row crops precisely. A small offset could mean that the weed is out of reach for the nozzles, leaving the weed untreated. This motivates the research in this paper to find a trajectory tracking controller that minimizes the tracking error.

Another aspect to consider is changing physical properties of the robot. For example, the weight of the robot will change as herbicide and fuel is consumed. To ensure satisfactory performance at all times, several adaptive approaches that update the controller gains continuously have been tested.

Unicycle-like robots are used extensively in all kinds of fields and numerous models and controllers have been described in publications. In this paper a nonlinear model proposed in [4] has been used for simulations. The same

Fig. 1. A picture of the prototype robot on a field test.

model was also used in [5] to develop an adaptive dynamic controller, which has been implemented and tested here but with a different adaptation law. In [6] an adaptive controller using adaptive backstepping is presented. [7] developed a model reference adaptive controller (MRAC) for the tracking problem, but only simulations were performed. A similar direct MRAC has been derived here and implemented on the robot for testing.

The most important contribution of this paper is the comparison of two different adaptive controllers implemented on the same robot. The author is not aware of any previous implementations of the MRAC controller presented here on a real robot.

Different approaches to row crop guidance systems has been thoroughly explored and reviewed in [8]. However, this paper focuses merely on tracking a smooth and well defined trajectory without considering how to obtain the trajectory. The results obtained should be applicable to most unicycle-like robots.

II. Mathematical Model

The model used for simulations and some of the controller designs in this paper was presented in [4]. It is given as

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \\ \dot{u} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} u\cos\psi - a\omega\sin\psi \\ u\sin\psi + a\omega\cos\psi \\ \omega \\ \frac{\theta_3}{\theta_1}\omega^2 - \frac{\theta_4}{\theta_1}u \\ -\frac{\theta_5}{\theta_2}u\omega - \frac{\theta_6}{\theta_2}\omega \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{1}{\theta_1} & 0 \\ 0 & \frac{1}{\theta_2} \end{bmatrix} \begin{bmatrix} u_{ref} \\ \omega_{ref} \end{bmatrix} + \begin{bmatrix} \delta_x \\ \delta_y \\ 0 \\ \bar{\delta}_u \\ \bar{\delta}_\omega \end{bmatrix} \tag{1}$$

where $\boldsymbol{h} = \begin{bmatrix} x & y \end{bmatrix}^T$ is the position, ψ is the heading angle, u is forward velocity, ω is angular velocity and a is the distance

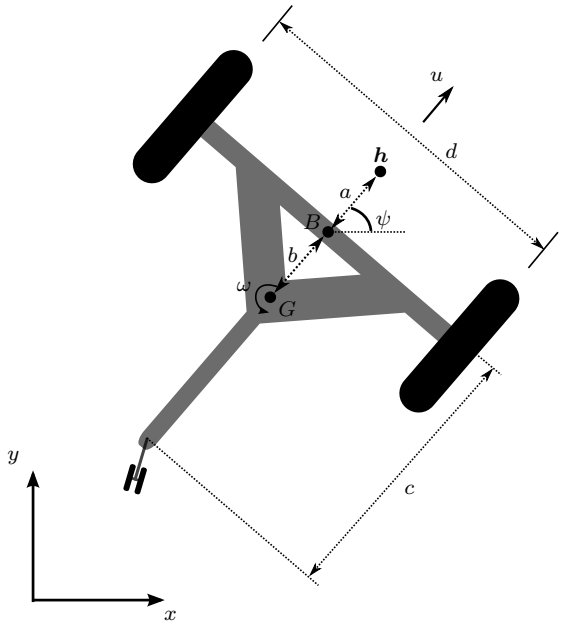


Fig. 2. Drawing of unicycle-like robot with centered rear mounted caster wheel, similar to the robot all tests were performed on.

from center of wheel axis to $\boldsymbol{h}$ as shown in Fig. 2. Motor inputs are given as velocities instead of torque values, which means that the motor controller is assumed to have a PID controller or similar. $\boldsymbol{\theta}$ is a collection of physical parameters derived in [4] and included here for reference:

$$\begin{aligned}
\theta_1 &= \left(\frac{R_a}{k_a}(mR_t r + 2I_e) + 2rk_{DT}\right)/(2rk_{PT}) \\
\theta_2 &= \left(\frac{R_a}{k_a}(I_e d^2 + 2R_t r(I_z + mb^2)) + 2rdk_{DR}\right)/(2rdk_{PR}) \\
\theta_3 &= \frac{R_a}{k_a} mbR_t/(2k_{PT}) \\
\theta_4 &= \frac{R_a}{k_a}\left(\frac{k_a k_b}{R_a} + B_e\right)/(rk_{PT}) + 1 \\
\theta_5 &= \frac{R_a}{k_a} mbR_t/(dk_{PR}) \\
\theta_6 &= \frac{R_a}{k_a}\left(\frac{k_a k_b}{R_a} + B_e\right) d/(2rk_{PR}) + 1
\end{aligned} \tag{2}$$

where R_a is motor resistance, k_a motor torque multiplied by gear ratio, k_b motor voltage multiplied by gear ratio, r wheel radius, I_e motor moment of inertia, B_e motor viscous friction coefficient, $k_{PT}, k_{DT}, k_{PR}, k_{DR}$ are PID motor controller gains, I_z moment of inertia about vertical axis at center of mass, m mass.

$$\boldsymbol{\delta} = \begin{bmatrix}\delta_x & \delta_y & 0 & \bar{\delta}_u & \bar{\delta}_\omega\end{bmatrix}^T \tag{3}$$

represent the uncertainties of the system caused by wheel slips and forces exerted by the caster wheel. For the purpose of this paper it has been assumed that $\boldsymbol{\delta} = 0$.

III. Controller Design

In many cases unicycle-like robots operate at low speeds and often inhibit low moment of inertia. In other words, the dynamics of u and ω are so fast that in many cases one may simplify $u \approx u_{ref},\ \omega \approx \omega_{ref}$ and only study the kinematic model given by

$$\begin{bmatrix}\dot{x} \\ \dot{y} \\ \dot{\psi}\end{bmatrix} = \begin{bmatrix}u_{ref}\cos\psi - a\omega_{ref}\sin\psi \\ u_{ref}\sin\psi + a\omega_{ref}\cos\psi \\ \omega_{ref}\end{bmatrix} \tag{4}$$

For larger robots operating at higher speeds the dynamics cannot simply be ignored. In the next sections, various controller designs are considered.

A. Trajectory Tracking Controller

Let $\boldsymbol{h_d}(t) = \begin{bmatrix}x_d(t) & y_d(t)\end{bmatrix}^T$ denote the time varying reference trajectory for the robot. Only the reference position is considered, he the The tracking error is defined as $\tilde{\boldsymbol{h}} = \begin{bmatrix}x_d - x & y_d - y\end{bmatrix}^T$. The kinematics from (1) may be written as

$$\begin{bmatrix}\dot{x} \\ \dot{y} \\ \dot{\psi}\end{bmatrix} = \begin{bmatrix}\cos\psi & -a\sin\psi \\ \sin\psi & a\cos\psi \\ 0 & 1\end{bmatrix}\begin{bmatrix}u \\ \omega\end{bmatrix} \tag{5}$$

Recalling that $\boldsymbol{h} = [x\ \ y]^T$

$$\dot{h} = \begin{bmatrix}\dot{x} \\ \dot{y}\end{bmatrix} = \begin{bmatrix}\cos\psi & -a\sin\psi \\ \sin\psi & a\cos\psi\end{bmatrix}\begin{bmatrix}u \\ \omega\end{bmatrix} = \boldsymbol{A}\begin{bmatrix}u \\ \omega\end{bmatrix} \tag{6}$$

Multiplying (6) with the inverse of $\boldsymbol{A}$ gives

$$\begin{bmatrix}u \\ \omega\end{bmatrix} = \boldsymbol{A}^{-1}\begin{bmatrix}\dot{x} \\ \dot{y}\end{bmatrix} = \begin{bmatrix}\cos\psi & \sin\psi \\ -\frac{1}{a}\sin\psi & \frac{1}{a}\cos\psi\end{bmatrix}\begin{bmatrix}\dot{x} \\ \dot{y}\end{bmatrix} \tag{7}$$

A controller based on inverse kinematics is proposed in [5]:

$$\begin{bmatrix}u^c_{ref} \\ \omega^c_{ref}\end{bmatrix} = \begin{bmatrix}\cos\psi & \sin\psi \\ -\frac{1}{a}\sin\psi & \frac{1}{a}\cos\psi\end{bmatrix}\begin{bmatrix}\dot{x}_d + l_x\tanh\left(\frac{k_x}{l_x}\tilde{x}\right) \\ \dot{y}_d + l_y\tanh\left(\frac{k_y}{l_y}\tilde{y}\right)\end{bmatrix} \tag{8}$$

where $k_x, k_y > 0$ are controller gains and $l_x, l_y > 0$ are saturation constants. u^c_{ref} and ω^c_{ref} are desired forward and angular velocities, respectively. The controller is shown in [5] to have an asymptotically stable equilibrium at the origin $\tilde{\boldsymbol{h}} = \begin{bmatrix}0 & 0\end{bmatrix}^T$ under the assumption of $u = u^c_{ref}$ and $\omega = \omega^c_{ref}$. The name trajectory tracking controller and kinematic controller both refer to the same controller for the rest of this paper.

Note that it is not necessary to explicitly control the desired heading, as discussed in [5]. Due to the non-holonomic nature of a differentially wheeled mobile robot, the heading will be tangent to the trajectory path given small position errors. Any heading deviation from the tangent will cause a change in position errors, so proving stability for the position will be sufficient.

B. Dynamic Controller

The kinematic controller will work adequately as long as the dynamics of the system are fast enough, i.e. the assumption of $u \approx u^c_{ref}$ and $\omega \approx \omega^c_{ref}$ is reasonable. In cases where the dynamics are too slow to be ignored or high precision tracking

is required, the kinematic controller alone may no longer be sufficient.

Consider the dynamic part of (1)

$$\begin{bmatrix}\dot{u}\\ \dot{\omega}\end{bmatrix} = \begin{bmatrix}\frac{\theta_3}{\theta_1}\omega^2 - \frac{\theta_4}{\theta_1}u\\ -\frac{\theta_5}{\theta_2}u\omega - \frac{\theta_6}{\theta_2}\omega\end{bmatrix} + \begin{bmatrix}\frac{1}{\theta_1} & 0\\ 0 & \frac{1}{\theta_2}\end{bmatrix}\begin{bmatrix}u_{ref}\\ \omega_{ref}\end{bmatrix} \tag{9}$$

Rearranging gives

$$\begin{bmatrix}u_{ref}\\ \omega_{ref}\end{bmatrix} = \begin{bmatrix}\theta_1\dot{u} - \theta_3\omega^2 + \theta_4 u\\ \theta_2\dot{\omega} + \theta_5 u\omega + \theta_6\omega\end{bmatrix} \tag{10}$$

Which may be written as

$$\begin{bmatrix}u_{ref}\\ \omega_{ref}\end{bmatrix} = \begin{bmatrix}\theta_1 & 0\\ 0 & \theta_2\end{bmatrix}\begin{bmatrix}\dot{u}\\ \dot{\omega}\end{bmatrix} + \begin{bmatrix}0 & 0 & -\omega^2 & u & 0 & 0\\ 0 & 0 & 0 & 0 & u\omega & \omega\end{bmatrix}\boldsymbol{\theta} \tag{11}$$

Motivated by the inverse dynamics in (11), [5] proposes the controller given as

$$\begin{bmatrix}u_{ref}\\ \omega_{ref}\end{bmatrix} = \begin{bmatrix}\theta_1 & 0\\ 0 & \theta_2\end{bmatrix}\begin{bmatrix}\sigma_1\\ \sigma_2\end{bmatrix} + \begin{bmatrix}0 & 0 & -\omega^2 & u & 0 & 0\\ 0 & 0 & 0 & 0 & u\omega & \omega\end{bmatrix}\boldsymbol{\theta} \tag{12}$$

where

$$\boldsymbol{\sigma} = \begin{bmatrix}\sigma_1\\ \sigma_2\end{bmatrix} = \begin{bmatrix}\dot{u}^c_{ref} + k_1\tilde{u}\\ \dot{\omega}^c_{ref} + k_2\tilde{\omega}\end{bmatrix}, \qquad \begin{matrix}\tilde{u} = u^c_{ref} - u\\ \tilde{\omega} = \omega^c_{ref} - \omega\end{matrix} \tag{13}$$

and $k_1, k_2 > 0$ are constant gains. In order to implement (12), the values of $\boldsymbol{\theta}$ must be known. Measuring or otherwise obtaining the parameters needed to calculate $\boldsymbol{\theta}$ may prove hard, thus the need to estimate $\boldsymbol{\theta}$ becomes a necessity. Replacing $\boldsymbol{\theta}$ with the estimate $\hat{\boldsymbol{\theta}}$ in (12) gives

$$\boldsymbol{\nu_{ref}} = \hat{\boldsymbol{D}}\boldsymbol{\sigma} + \boldsymbol{E}\hat{\boldsymbol{\theta}} \tag{14}$$

where

$$\begin{aligned}&\boldsymbol{\nu_{ref}} = \begin{bmatrix}u_{ref}\\ \omega_{ref}\end{bmatrix},\ \boldsymbol{E} = \begin{bmatrix}0 & 0 & -\omega^2 & u & 0 & 0\\ 0 & 0 & 0 & 0 & u\omega & \omega\end{bmatrix},\\ &\hat{\boldsymbol{D}} = \begin{bmatrix}\hat{\theta}_1 & 0\\ 0 & \hat{\theta}_2\end{bmatrix},\ \hat{\boldsymbol{\theta}} = \begin{bmatrix}\hat{\theta}_1 & \hat{\theta}_2 & \hat{\theta}_3 & \hat{\theta}_4 & \hat{\theta}_5 & \hat{\theta}_6\end{bmatrix}^T\end{aligned} \tag{15}$$

Following is a stability analysis similar to what was done in [5]. (11) may be written as

$$\boldsymbol{\nu_{ref}} = \boldsymbol{D}\dot{\boldsymbol{\nu}} + \boldsymbol{E}\boldsymbol{\theta} \tag{16}$$

Similarly, (14) is written

$$\boldsymbol{\nu_{ref}} = \hat{\boldsymbol{D}}\boldsymbol{\sigma} + \boldsymbol{E}\hat{\boldsymbol{\theta}} = \boldsymbol{G}\boldsymbol{\theta} - \boldsymbol{G}\tilde{\boldsymbol{\theta}} = \boldsymbol{D}\boldsymbol{\sigma} + \boldsymbol{E}\boldsymbol{\theta} - \boldsymbol{G}\tilde{\boldsymbol{\theta}} \tag{17}$$

where $\tilde{\boldsymbol{\theta}} = \boldsymbol{\theta} - \hat{\boldsymbol{\theta}}$ and

$$\boldsymbol{G} = \begin{bmatrix}\sigma_1 & 0 & -\omega^2 & u & 0 & 0\\ 0 & \sigma_2 & 0 & 0 & u\omega & \omega\end{bmatrix},\ \boldsymbol{D} = \begin{bmatrix}\theta_1 & 0\\ 0 & \theta_2\end{bmatrix} \tag{18}$$

(13) may be written as

$$\boldsymbol{\sigma} = \begin{bmatrix}\sigma_1\\ \sigma_2\end{bmatrix} = \dot{\boldsymbol{\nu}}^{\boldsymbol{c}}_{\boldsymbol{ref}} + \boldsymbol{K}\tilde{\boldsymbol{\nu}} \tag{19}$$

where $\boldsymbol{K} = \mathrm{diag}(k_1, k_2)$. Combining (16), (17) and (19)

$$\boldsymbol{D}\dot{\boldsymbol{\nu}} + \boldsymbol{E}\boldsymbol{\theta} = \boldsymbol{D}\dot{\boldsymbol{\nu}}^{\boldsymbol{c}}_{\boldsymbol{ref}} + \boldsymbol{D}\boldsymbol{K}\tilde{\boldsymbol{\nu}} + \boldsymbol{E}\boldsymbol{\theta} - \boldsymbol{G}\tilde{\boldsymbol{\theta}} \tag{20}$$

$$\dot{\tilde{\boldsymbol{\nu}}} = -\boldsymbol{K}\tilde{\boldsymbol{\nu}} + \boldsymbol{D}^{-1}\boldsymbol{G}\tilde{\boldsymbol{\theta}} \tag{21}$$

where $\dot{\tilde{\boldsymbol{\nu}}} = \dot{\boldsymbol{\nu}}^{\boldsymbol{c}}_{\boldsymbol{ref}} - \dot{\boldsymbol{\nu}}$ describes the error dynamics of the system. For this analysis, $\boldsymbol{\theta}$ is considered known, i.e. $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}$, reducing (21) to

$$\dot{\tilde{\boldsymbol{\nu}}} = -\boldsymbol{K}\tilde{\boldsymbol{\nu}} \tag{22}$$

Consider the following Lyapunov-like function

$$V = \frac{1}{2}\tilde{\boldsymbol{\nu}}^T\boldsymbol{P}\tilde{\boldsymbol{\nu}} \tag{23}$$

where $\boldsymbol{P} = \boldsymbol{P}^T > \boldsymbol{0}$. Differentiating (23) along the solution of (22) gives

$$\dot{V} = -\tilde{\boldsymbol{\nu}}^T\boldsymbol{P}\boldsymbol{K}\tilde{\boldsymbol{\nu}} < 0 \ \ \forall \ \ \tilde{\boldsymbol{\nu}} \neq 0 \tag{24}$$

Which means that $\dot{V}$ is negative definite and global asymptotic stability can be concluded.

C. On-line Parameter Estimation

The dynamic controller given by (14) needs a good estimate $\hat{\boldsymbol{\theta}}$ in order to perform well. One approach to estimate $\hat{\boldsymbol{\theta}}$ is to log a test run with sufficiently excited input signal and use an off-line system identification technique, e.g. least-squares method. Another approach is to estimate $\hat{\boldsymbol{\theta}}$ on-line using an adaptation law $\dot{\hat{\boldsymbol{\theta}}}$. This section shows the derivation of $\dot{\hat{\boldsymbol{\theta}}}$ using the gradient method, which is motivated by the minimization of a cost function.

Consider (10) written on the form

$$\boldsymbol{\nu_{ref}} = \boldsymbol{\varphi}^T\boldsymbol{\theta} = \begin{bmatrix}\dot{u} & 0 & -\omega^2 & u & 0 & 0\\ 0 & \dot{\omega} & 0 & 0 & u\omega & \omega\end{bmatrix}\boldsymbol{\theta} \tag{25}$$

where $\boldsymbol{\nu_{ref}} = [u_{ref}\ \ \omega_{ref}]^T$. Filtering both sides gives

$$\frac{\boldsymbol{\nu_{ref}}}{\Lambda(s)} = \frac{\boldsymbol{\varphi}^T\boldsymbol{\theta}}{\Lambda(s)} = \frac{1}{\Lambda(s)}\begin{bmatrix}su & 0 & -\omega^2 & u & 0 & 0\\ 0 & s\omega & 0 & 0 & u\omega & \omega\end{bmatrix}\boldsymbol{\theta} \tag{26}$$

Which may be written as the parametric model

$$\boldsymbol{z} = \boldsymbol{\Phi}^T\boldsymbol{\theta} \tag{27}$$

where $z = \frac{\boldsymbol{\nu_{ref}}}{\Lambda(s)}$, $\boldsymbol{\Phi}^T = \frac{\boldsymbol{\varphi}^T}{\Lambda(s)}$ and $\Lambda(s)$ is chosen to be a Hurwitz polynomial of degree one, e.g. $\Lambda(s) = s + 1$. Note that $\boldsymbol{z}$ and $\boldsymbol{\Phi}$ are available measurements, while $\boldsymbol{\theta}$ is unknown. An estimate of $\boldsymbol{z}$ denoted $\hat{\boldsymbol{z}}$ is generated as

$$\hat{\boldsymbol{z}} = \boldsymbol{\Phi}^T\hat{\boldsymbol{\theta}} \tag{28}$$

where $\hat{\boldsymbol{\theta}}$ is the currently best estimate of $\boldsymbol{\theta}$. A normalized estimation error is defined as

$$\boldsymbol{\epsilon} = (\boldsymbol{M}^T\boldsymbol{M})^{-1}(\boldsymbol{z} - \hat{\boldsymbol{z}}) = (\boldsymbol{M}^T\boldsymbol{M})^{-1}(\boldsymbol{z} - \boldsymbol{\Phi}^T\hat{\boldsymbol{\theta}}) \tag{29}$$

where $\boldsymbol{M}^T\boldsymbol{M} = \boldsymbol{I} + \boldsymbol{N_s}^T\boldsymbol{N_s}$ is a diagonal matrix that normalizes the estimation error, and $\boldsymbol{N_s}^T\boldsymbol{N_s}$ is another diagonal matrix for design of the normalized signal. The reason for this normalization is to ensure boundedness, i.e.

$$\boldsymbol{\Phi}\boldsymbol{M}^{-1} \in \mathcal{L}_\infty \tag{30}$$

If $\boldsymbol{\Phi} \in \mathcal{L}_\infty$, then $\boldsymbol{M} = \boldsymbol{I}$ is sufficient. If it is not, choosing

$$\boldsymbol{M}^T\boldsymbol{M} = \boldsymbol{I} + \boldsymbol{\Phi}^T\boldsymbol{\Phi} \tag{31}$$

will ensure (30) is satisfied [9, p. 172]. An instantaneous cost function $\boldsymbol{J}(\hat{\boldsymbol{\theta}})$ is defined as

$$\boldsymbol{J}(\hat{\boldsymbol{\theta}}) = \frac{1}{2}\boldsymbol{\epsilon}^T\boldsymbol{M}^T\boldsymbol{M}\boldsymbol{\epsilon} = \frac{1}{2}(\boldsymbol{z} - \boldsymbol{\Phi}^T\hat{\boldsymbol{\theta}})^T(\boldsymbol{M}^T\boldsymbol{M})^{-1}(\boldsymbol{z} - \boldsymbol{\Phi}^T\hat{\boldsymbol{\theta}}) \tag{32}$$

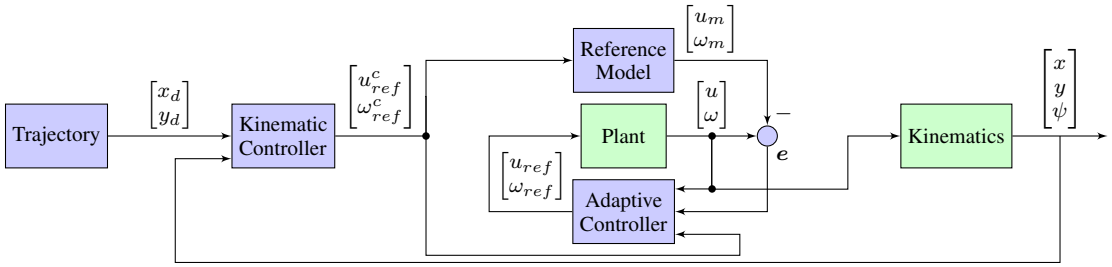


Fig. 3. Block diagram of the model reference adaptive controller.

The gradient of (32) is

$$\Delta \boldsymbol{J}(\hat{\boldsymbol{\theta}}) = -\boldsymbol{\Phi}(\boldsymbol{M}^T\boldsymbol{M})^{-1}(\boldsymbol{z} - \boldsymbol{\Phi}^T\hat{\boldsymbol{\theta}}) = -\boldsymbol{\Phi}\boldsymbol{\epsilon} \tag{33}$$

Motivated by this, the following adaptation law for generating $\hat{\boldsymbol{\theta}}(t)$ is proposed

$$\dot{\hat{\boldsymbol{\theta}}} = -\boldsymbol{\Gamma}\Delta \boldsymbol{J}(\hat{\boldsymbol{\theta}}) = \boldsymbol{\Gamma}\boldsymbol{\Phi}\boldsymbol{\epsilon} \tag{34}$$

where $\boldsymbol{\Gamma} = \boldsymbol{\Gamma}^T >$ is a diagonal gain matrix. According to [9, p. 175], (34) ensures that

1) $\hat{\boldsymbol{\theta}}, \boldsymbol{\epsilon} \in \mathcal{L}_\infty$
2) $\boldsymbol{\epsilon}, \boldsymbol{N_s}^T\boldsymbol{\epsilon}, \dot{\hat{\boldsymbol{\theta}}} \in \mathcal{L}_\infty$

independent of the boundedness properties of $\boldsymbol{\Phi}$. In other words, both parameters and estimation errors should remain bounded. It does not, however, ensure that $\tilde{\boldsymbol{\theta}}(t) = \boldsymbol{\theta}(t) - \hat{\boldsymbol{\theta}}(t) \to 0$ as $t \to \infty$. To ensure that the parameters $\hat{\boldsymbol{\theta}}$ do in fact converge to their actual value $\boldsymbol{\theta}$, $\boldsymbol{\Phi}$ must be persistently excited (PE), i.e., it satisfies [9, p. 254]

$$\alpha_1 \boldsymbol{I} \geq \frac{1}{T_0}\int_t^{t+T_0} \boldsymbol{\Phi}(\tau)\boldsymbol{\Phi}^T(\tau)\,\mathrm{d}\tau \geq \alpha_0 \boldsymbol{I}, \quad \forall t \geq 0 \tag{35}$$

for some $T_0, \alpha_0, \alpha_1 \geq 0$. It is in general difficult to show that $\boldsymbol{\Phi}$ is PE for an input signal $\boldsymbol{\nu_{ref}}$, and especially in a case like this where $\boldsymbol{\Phi}$ has some nonlinear elements.

D. Adaptive Dynamic Controller

The results from section III-B and section III-C may be combined to form an adaptive dynamic controller. The on-line parameter estimation operates independently from the dynamic controller and vice versa, making it a modular design. This may prove beneficial in cases where parameter estimation is only needed parts of the time, or if it is desirable to run parameter estimation without running the dynamic controller. The control laws are given by

$$\boldsymbol{\nu_{ref}} = \hat{\boldsymbol{D}}\boldsymbol{\sigma} + \boldsymbol{E}\hat{\boldsymbol{\theta}}, \quad \dot{\hat{\boldsymbol{\theta}}} = \boldsymbol{\Gamma}\boldsymbol{\Phi}\boldsymbol{\epsilon} \tag{36}$$

where the notation is the same as in section III-B and section III-C.

E. Direct Model Reference Adaptive Controller

In this section, a simple direct Model Reference Adaptive Controller (MRAC) scheme as shown in Fig. 3 is derived. The concept is to design a model of similar structure to the plant (robot), let the tracking reference be an input to the model, and make the output of the plant track the output of the model. For the direct MRAC approach, this is made possible by developing adaptation laws for the controller gains directly without having to identify actual system parameters.

Consider a simplified, linear model of the dynamics of (1) given by

$$\begin{bmatrix} \dot{u} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} au + bu_{ref} \\ c\omega + d\omega_{ref} \end{bmatrix} \tag{37}$$

where a, b, c, d are unknown system parameters (not the same as those introduced in Fig. 2). In this case, u and ω are considered decoupled and will be analyzed separately. A reference model u_m for u is chosen to be

$$\dot{u}_m = -a_m u + b_m u^c_{ref} \tag{38}$$

Laplace transforming (37) and (38) gives

$$u = \frac{b}{s-a}u_{ref}, \quad u_m = \frac{b_m}{s+a_m}u^c_{ref} \tag{39}$$

The following control law is proposed

$$u_{ref} = -k^*_u u + l^*_u u^c_{ref} \tag{40}$$

Inserting (40) into (39) gives

$$u = \frac{bl^*_u}{s - a + bk^*_u}u^c_{ref}, \quad u_m = \frac{b_m}{s+a_m}u^c_{ref} \tag{41}$$

It is desirable to make the transfer functions of (41) equal. Choosing

$$l^*_u = \frac{b_m}{b}, \quad k^*_u = \frac{a + a_m}{b} \tag{42}$$

ensures equal transfer functions. However, it is not possible to implement since the values of a and b are unknown. Instead of using the control law (40), a control law using estimates of k^*_u and l^*_u is proposed

$$u_{ref} = -k_u(t)u + l_u(t)u^c_{ref} \tag{43}$$

where $k_u(t)$ and $l_u(t)$ are the currently best estimates of k^*_u and l^*_u, respectively. Adding and subtracting $b(-k^*_u u + l^*_u u^c_{ref})$ to $\dot{u}$ yields

$$\dot{u} = au + bu_{ref} + b(-k^*_u u + l^*_u r) - b(-k^*_u u + l^*_u r) \tag{44}$$

which, after combining with (42), may be written as

$$\dot{u} = -a_m u + b_m u^c_{ref} u + b(k^*_u u - l^*_u u^c_{ref} + u_{ref}) \tag{45}$$

Laplace transforming (45) gives

$$u = \underbrace{\frac{b_m}{s+a_m}u^c_{ref}}_{=u_m} + \frac{b}{s+a_m}(k^*_u u - l^*_u u^c_{ref} + u_{ref}) \tag{46}$$

Define the tracking error $e_u = u - u_m$ to obtain

$$e_u = \frac{b}{s + a_m}(k_u^* u - l_u^* u_{ref}^c + u_{ref}) \tag{47}$$

Since k_u^* and l_u^* are unknown, our best estimate of the tracking error $\hat{e}_u$ is

$$\hat{e}_u = \frac{b}{s + a_m}(k_u(t) u - l_u(t) u_{ref}^c + u_{ref}) \tag{48}$$

Inserting u_{ref} from (43) into (48) simply gives $\hat{e}_u = 0$, i.e. the estimated tracking error is zero. Note that the estimation error $\epsilon_u = e_u - \hat{e}_u = e_u = u - u_m$ is equal to the tracking error e_u. Combining (43) and (47) while defining the gain parameter estimation errors $\tilde{k}_u(t) = k_u(t) - k_u^*$, $\tilde{l}_u(t) = l_u(t) - l_u^*$ gives

$$e_u = \frac{b}{s + a_m}(-\tilde{k}_u u + \tilde{l}_u u_{ref}^c) \tag{49}$$

$$\dot{e}_u = -a_m e_u + b(-\tilde{k}_u u + \tilde{l}_u u_{ref}^c) \tag{50}$$

Consider the Lyapunov-like function

$$V = \frac{1}{2} e_u^2 + \frac{|b|}{2\gamma_1}\tilde{k}_u^2 + \frac{|b|}{2\gamma_2}\tilde{l}_u^2 \tag{51}$$

with $\gamma_1, \gamma_2 > 0$. Differentiating (51) along the solution of (49) gives

$$\begin{aligned} \dot{V} = &- a_m e_u^2 + |b|\tilde{k}_u \left(-e_u u \operatorname{sgn}(b) + \frac{1}{\gamma_1}\dot{k} \right) \\ &+ |b|\tilde{l}_u \left(e_u u_{ref}^c \operatorname{sgn}(b) + \frac{1}{\gamma_2}\dot{l} \right) \end{aligned} \tag{52}$$

Choosing

$$\dot{k}_u = \gamma_1 e_u u \operatorname{sgn}(b), \quad \dot{l}_u = -\gamma_2 e_u u_{ref}^c \operatorname{sgn}(b) \tag{53}$$

ensures

$$\dot{V} = -a_m e_u^2 \leq 0 \tag{54}$$

Thus it is shown that $\dot{V}$ is negative semi definite, V has an upper bound $V(0)$ and bounded below by zero, i.e. $0 \leq V(t) \leq V(0)$. From the boundedness of $V(t)$ and (51), it is clear that $e_u, \tilde{k}_u, \tilde{l}_u \in \mathcal{L}_\infty$. u_{ref}^c, the output of the kinematic controller (8), is bounded, so $u_{ref}^c \in \mathcal{L}_\infty$. The transfer functions of (41) are in $\mathcal{L}_1$, and it follows from [9, p. 80] that $u, u_m \in \mathcal{L}_\infty$. This means that all signals of (49) are bounded and $\dot{e}_u \in \mathcal{L}_\infty$. From [9, p. 74] it follows that since $V(t)$ is bounded from below and non-increasing, it has a finite limit as $t \to \infty$, denoted V_∞. It can also be seen that

$$\begin{aligned} ||e_u||_2 &= \left(\int_0^\infty e_u^2(\tau)\mathrm{d}\tau \right)^{1/2} = \left(\int_0^\infty -\frac{1}{a_m}\dot{V}(\tau)\mathrm{d}\tau \right)^{1/2} \\ &= \left(\frac{1}{a_m}(V(0) - V_\infty) \right)^{1/2} \end{aligned} \tag{55}$$

which is clearly bounded, so that $e_u \in \mathcal{L}_2$. Finally, [9, p. 80] shows that since $\dot{e}_u, e_u \in \mathcal{L}_\infty$ and $e_u \in \mathcal{L}_2$, then $e_u(t) \to 0$ as $t \to \infty$.

The results obtained show that the tracking objective of making the output of the plant $u(t)$ track the output of the reference model $u_m(t)$ is achieved. It does not, however, guarantee that $k_u(t), l_u(t) \to k_u^*, l_u^*$ as $t \to \infty$, i.e. the poles of the plant may differ from those of the reference model. This should be of less concern, since $k_u(t), l_u(t)$ are bounded and the true values of k_u^*, l_u^* are not of any real importance.

In a very similar manner the same results are found for ω. A summary of the control laws are given in Table I.

A modification that was done to provide for a more robust implementation was to add a small feedback loop to (53) to get

$$\dot{k}_u = \gamma_1 e_u u \operatorname{sgn}(b) - \alpha k_u, \quad \dot{l}_u = -\gamma_2 e_u u_{ref}^c \operatorname{sgn}(b) - \beta l_u \tag{56}$$

where $0 < \alpha << 1$ and $0 < \beta << 1$.

TABLE I. Control Laws for The MRAC

Plant	Reference Model	Control Law
$u = \frac{b}{s-a} u_{ref}$	$u_m = \frac{b_m}{s+a_m} u_{ref}^c$	$u_{ref} = -k_u(t)u + l_u(t)u_{ref}^c$, $\dot{k}_u = \gamma_1 e_u u \operatorname{sgn}(b)$ $\dot{l}_u = -\gamma_2 e_u u_{ref}^c \operatorname{sgn}(b)$ $e_u = u - u_m$
$\omega = \frac{d}{s-c}\omega_{ref}$	$\omega_m = \frac{d_m}{s+c_m}\omega_{ref}^c$	$\omega_{ref} = -k_\omega(t)\omega + l_\omega(t)\omega_{ref}^c$, $\dot{k}_\omega = \gamma_3 e_\omega \omega \operatorname{sgn}(d)$ $\dot{l}_\omega = -\gamma_4 e_\omega \omega_{ref}^c \operatorname{sgn}(d)$ $e_\omega = \omega - \omega_m$

IV. Simulations And Real Runs

Simulations were performed using Matlab/Simulink, while real runs were performed on the robot shown in Fig. 4. The dimensions of the robot are approximately $c = 1.1\,\mathrm{m}$ and $d = 1.0\,\mathrm{m}$ using the notation from Fig. 2. The robot is running Robot Operating System (ROS) and all controllers were implemented in C++. The motor controller has a low level PID controller that uses individual motor velocities as setpoints, and motor acceleration can be saturated to ensure slower dynamics. Without limits on acceleration the dynamics were so fast that all controllers had equal performance. For both simulations and tests on the real robot the following figure eight trajectory was used:

$$\begin{aligned} x_d(t) &= r_e \sin(2\omega_e t) \\ y_d(t) &= r_e(\cos(\omega_e t) - 1) \end{aligned} \tag{57}$$

For both simulations and real runs $r_e = 0.6\,\mathrm{m}$ and $\omega_e = 0.3\,\mathrm{rad/s}$. The distance from wheel axle to $\boldsymbol{h}$ was chosen to be $a = 0.10\,\mathrm{m}$. The simulated system uses $\boldsymbol{\theta} = \begin{bmatrix} 1.0 & 0.4 & 0.2 & 1.1 & 0.2 & 0.9 \end{bmatrix}^T$ and initial estimates $\hat{\boldsymbol{\theta}}_\mathbf{0} = \begin{bmatrix} 1.0 & 1.0 & 0 & 1.0 & 0 & 1.0 \end{bmatrix}^T$ are used for the dynamic and adaptive dynamic controllers. The on-line parameter estimation method was tested in simulations where $\boldsymbol{\theta}$ is known to ensure that the estimated $\hat{\boldsymbol{\theta}}$ converges to its actual value $\boldsymbol{\theta}$. It was found that $\hat{\boldsymbol{\theta}}$ does indeed converge correctly while attempting to track the figure eight, which means that the input signal is sufficiently excited to ensure convergence.

A comparison of all controllers is shown in Fig. 8. The estimated parameters during a real run are shown in Fig. 5 and Fig. 6. Controller inputs during a real run are shown in Fig. 7. It is clear from Fig. 8 (a) and (e) that the kinematic controller alone does not provide sufficient performance in this case. The dynamic controller shows good performance given

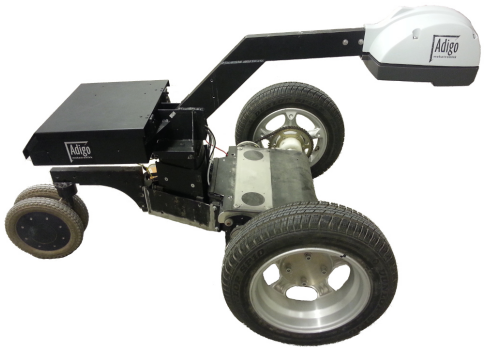


Fig. 4. A picture of the robot used for the tests. The robot uses a rear caster wheel.

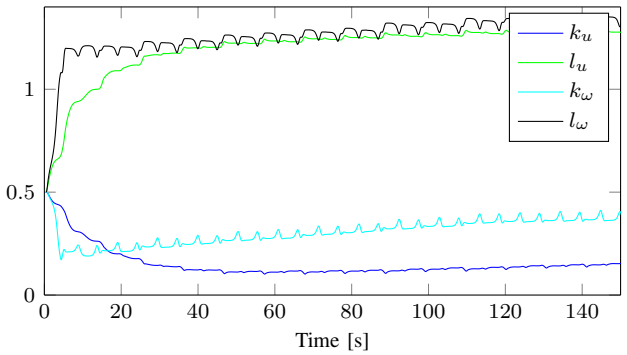


Fig. 5. Controller gains for the MRAC during a real run. All controller gains have initial values of 0.5.

fairly accurate estimates $\hat{\boldsymbol{\theta}}$. The adaptive dynamic controller appears to be able to improve upon the performance of the dynamic controller as $\hat{\boldsymbol{\theta}}$ adapts (shown in Fig. 6).

In Fig. 8 (i) and (j) the motor acceleration saturation limit was increased to make the system a bit faster. It is interesting that the MRAC improves greatly when the dynamics are faster, while the adaptive dynamic controller has almost identical performance to the case with slower dynamics.

V. Conclusion

Two different adaptive dynamic controllers for tracking a trajectory were implemented on a differentially wheeled robot and compared with non-adaptive kinematic and dynamic controllers. The MRAC configuration, which the author has been unable to find previous papers presenting real implementations of, delivered the best performance of all controllers on a system with fairly slow dynamics, while the other adaptive dynamic controller had equal or better performance on a very slow system.

References

[1] F. Urdal, T. Utstumo, S. A. Ellingsen, J. K. Vatne, and T. Gravdahl, "Design and control of precision drop-on-demand herbicide application in agricultural robotics," in *The 13th international Conference on Control, Automation, Robotics and Vision, Singapore*. IEEE, 2014.

[2] T. Utstumo, T. Berge, and J. T. Gravdahl, "Non-linear model predictive control for constrained robot navigation in row crops," *in the Proceedings of the 2015 IEEE International Conference on Industrial Technology (ICIT 2015), Seville, Spain*, March 17-19 2015.

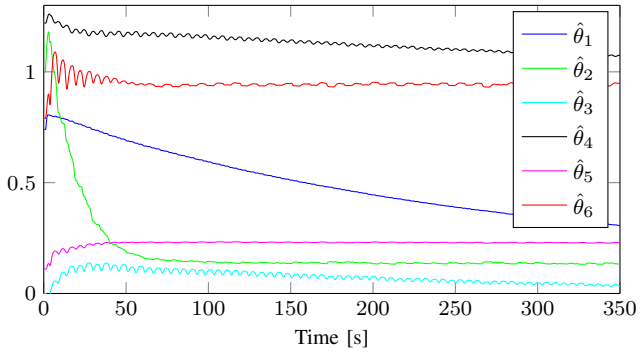


Fig. 6. Estimated system parameters during a real run with the adaptive dynamic controller with $\hat{\boldsymbol{\theta}}_\mathbf{0} = [0.74\ 1.00\ 0.00\ 1.22\ 0.11\ 0.79]^T$ and $\boldsymbol{\Gamma} = \text{diag}(1.0, 1.0, 1.0, 1.0, 1.0, 1.0)$.

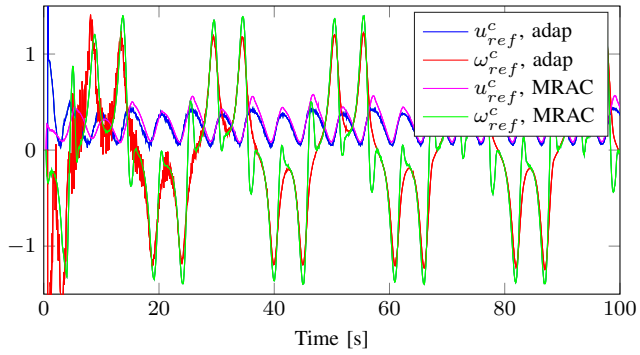


Fig. 7. Motor controller inputs from the MRAC and the adaptive dynamic controller during a test run with the slow dynamics. The inputs become smoother as the parameters adapt.

[3] T. Utstumo and J. T. Gravdahl, "Implementation and comparison of attitude estimation methods for agricultural robotics," *Agricontrol. Vol. 4. No. 1. 2013*, pp. 52–57, 2013.

[4] C. De La Cruz and R. Carelli, "Dynamic Modeling and Centralized Formation Control of Mobile Robots," *IECON 2006 - 32nd Annual Conference on IEEE Industrial Electronics*, pp. 3880–3885, Nov. 2006. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4153091

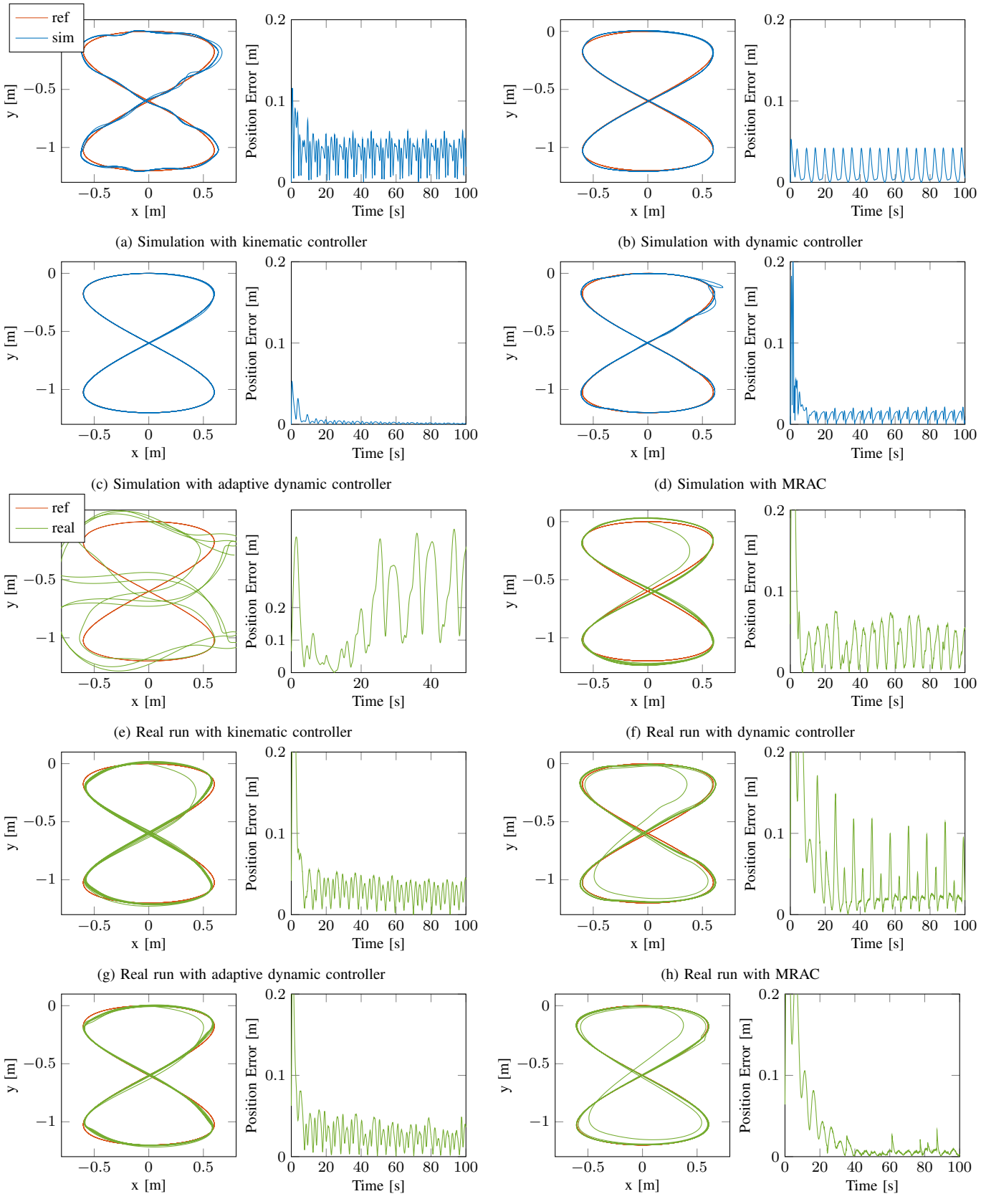
[5] F. N. Martins, W. C. Celeste, R. Carelli, M. Sarcinelli-Filho, and T. F. Bastos-Filho, "An adaptive dynamic controller for autonomous mobile robot trajectory tracking," *Control Engineering Practice*, vol. 16, no. 11, pp. 1354–1363, Nov. 2008. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0967066108000373

[6] T. Fukao, H. Nakagawa, and N. Adachi, "Adaptive tracking control of a nonholonomic mobile robot," *Robotics and Automation, IEEE Transactions on*, vol. 16, no. 5, pp. 609–615, Oct 2000.

[7] E. Canigur and M. Ozkan, "Model reference adaptive control of a nonholonomic wheeled mobile robot for trajectory tracking," in *Innovations in Intelligent Systems and Applications (INISTA), 2012 International Symposium on*, July 2012, pp. 1–5.

[8] F. Dong, W. Heinemann, and R. Kasper, "Development of a row guidance system for an autonomous robot for white asparagus harvesting," *Computers and Electronics in Agriculture*, vol. 79, no. 2, pp. 216 – 225, 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0168169911002304

[9] P. Ioannou and J. Sun, *Robust Adaptive Control*. Dover Publications, 2013. [Online]. Available: https://books.google.no/books?id=ffavAAAAQBAJ



(a) Simulation with kinematic controller

(b) Simulation with dynamic controller

(c) Simulation with adaptive dynamic controller

(d) Simulation with MRAC

(e) Real run with kinematic controller

(f) Real run with dynamic controller

(g) Real run with adaptive dynamic controller

(h) Real run with MRAC

(i) Real run with adaptive dynamic controller on a faster system

(j) Real run with MRAC on a faster system

Fig. 8. Comparison of all the controllers with a relatively slow system and two selected runs on a faster system.