Noise sensitivity of selected kinematic path following controllers for a unicycle

Article *in* Bulletin of the Polish Academy of Sciences, Technical Sciences · March 2014

DOI: 10.2478/bpasts-2014-0001

2 authors, including:

Urszula Libal
KTH Royal Institute of Technology
24 PUBLICATIONS **70** CITATIONS

BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES, Vol. 62, No. 1, 2014
DOI: 10.2478/bpasts-2014-0001

Noise sensitivity of selected kinematic path following controllers for a unicycle

U. LIBAL and J. PŁASKONKA*

Institute of Computer Engineering, Control and Robotics, Wroclaw University of Technology,
11/17 Janiszewski St., 50-372 Wroclaw, Poland

Abstract. In the paper a path following problem for a wheeled mobile robot of (2,0) type has been considered. The kinematic model of the robot was derived with respect to the Serret-Frenet frame. Two kinematic control algorithms – Samson and Morin-Samson – have been tested taking into account their sensitivity to a white noise with a zero mean appearing in the one of state variables. The properties of path following errors have been analysed using statistical techniques. The conclusions related to an acceptable level of noise and a range of applicability of the presented algorithms have been reached.

Key words: wheeled mobile robots, path following, Serret-Frenet frame, model-based control, nonlinear control systems, nonholonomic constraints, probabilistic robotics.

1. Introduction

The problem of designing a control law for nonholonomic systems has been deeply investigated in the literature, see e.g. [1–3]. Among the nonholonomic robots one can distinguish wheeled mobile robots [4]. As mobile robots are more and more often used in many different applications, just to mention mobile platforms for planetary explorations, cars that travel by themselves and autonomous vacuum cleaners, there is still a need for a research related to effective control algorithms for such robotic objects. One can distinguish three basic types of tasks realized by mobile robots:

- point stabilization – the robot should be stabilized at a given target point,
- trajectory tracking – the vehicle has to track a time-parametrized curve,
- path following – the robot has to follow a path which is parametrized by a curvilinear distance from a fixed point.

In the paper we focus on a path following task.

There are many path following algorithms published in the literature which use model-based motion planning techniques. The path following of the mobile platforms with nonholonomic constraints can be realized e.g. by the Pomet theorem [5], what was presented in [6], or using algorithms dedicated to particular types of the platforms. The important advantage of the Pomet algorithm is that it can be applied to many nonholonomic systems, however it gives oscillatory transient states with slow convergence rate. The example of the algorithm dedicated to a certain type of mobile robots is the Samson algorithm [7] which guarantees an asymptotic path following for a unicycle – the wheeled mobile robot of (2,0) type.

The approach presented in [8] bases on controlling explicitly the rate of progression of a "virtual target" to be tracked along the path. Such an approach overcomes the restriction imposed on the initial position of the robot in the case when the position of the virtual target is simply defined by the orthogonal projection of the actual vehicle on that path. The method proposed in [9] solves the problem of the bounded path curvature. It neither requires the computation of a projection of the robot position on the path, nor does it need to consider a moving virtual target to be tracked.

The another idea of developing control strategy in a path following task is to transform kinematic equations of the robot derived with respect to the Serret-Frenet frame into a canonical form called the chained form and then propose an algorithm for such a system. Morin and Samson applied that solution and introduced the algorithm which may be used e.g. for the path following of the unicycle, [10]. The same idea to derive the kinematic equations of a car-like robot in terms of path coordinates and then transform them into the chained form was presented in [11].

When designing new control strategies, one has to verify if the proposed method works properly. The evaluation of an algorithm could be done using a real robot or simulation experiments. In general the simulation approach should be faster and cheaper than experiments on a real robot. However it is not always obvious if lessons learnt during simulation experiments are relevant to real-life applications. The existing robots have to deal with the uncertainty that exists in the physical world and result from many different sources. Some of the researches provide only the simulations which do not take into account the robot's uncertainty. The question arises how such control strategies would cope in real-life applications. The problem of the validity of computer simulations in the

*e-mail: joanna.plaskonka@pwr.wroc.pl

process of developing control systems for autonomous robots was described in e.g. [12, 13]. The comprehensive overview of the methods used in robotics which focus on the uncertainty in robot perception and action could be found in [14].

One can use different methods to verify the effectiveness of designed control algorithms. Usually simulations or experiments on a real robot are performed. The majority of the simulation approaches are based on a deterministic model which may be insufficient and too simplified. In contrary, experiments on real robots might be more expensive and longer lasting than simulations. What is more, experiments allow to observe results for narrower range of parameters, due to a fixed robot construction, comparing to simulations. Thus that was a motivation to undertake research by simulation studies for probabilistic model.

The topic concerning the investigation of the impact of disturbances on the realization of the task by the robot is not so popular among the researches. In [15] the authors provided simple definition of robustness of asymptotic stabilizers with respect to modelling error. What is more, the conclusion that for chained form systems no continuous homogeneous stabilizer can be robust was made. The problem of global robust exponential stabilization of nonholonomic chained systems in the presence of sensor noise and external disturbances has been considered in [16]. It was shown that the problem is solvable by means of a simple hybrid control law. There are practical approaches known from the literature, see e.g. [17], which test theoretically designed algorithms by performing chosen motion tasks on a real robot, however the probabilistic analysis of the behaviour of tracking errors is omitted. The comparison of the algorithms adapted to a control of a three-link nonholonomic manipulator shown in [18] is based on the simulation results and takes into account, inter alia, the algorithms' robustness to measurements errors.

The aim of this paper is to verify the effectiveness of the selected kinematic path following controllers basing on simulation experiments in the situation when one of the state variables is noised. The presented research may be seen as a middle point between a theoretical design and a practical application on a real machine. To describe the courses of the path tracking errors some statistical techniques have been applied and because of that the presented paper could be of interest not only to robotics researchers but also to applied statisticians.

2. Description of a robot relative to a given path

Let us consider a wheeled mobile robot of (2,0) class moving on a plane. Such a robot with a path that has to be followed was shown in Fig. 1.

We make an assumption that the robot's wheels are nondeformable, they move without slip and as a result the nonholonomic constraints appear in the motion of the robot. The state variables of the mobile robot are described by a vector of coordinates $q = (x, y, \theta)^T$, where (x, y) denote the position of the point M, which is the robot's guidance point located in the middle of the wheel axle of the vehicle, and θ is a robot's chassis orientation with a respect to the inertial frame X_0Y_0. The kinematic model of the unicycle robot can be expressed by the equations

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{pmatrix} v \\ \omega \end{pmatrix} = G(q)\eta. \tag{1}$$

Symbols v and ω denote linear and angular velocities, respectively.

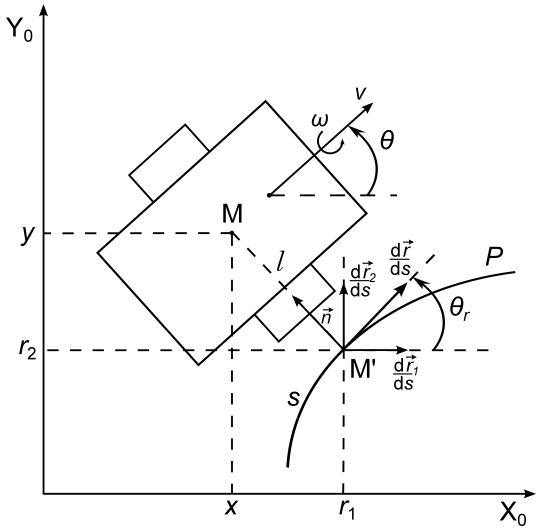

Fig. 1. The parameters of the unicycle and Serret-Frenet frame definition

2.1. Representation in a Serret-Frenet frame. To determine the position of the considered object relative to a desired path one may use different parametrizations. One of the most popular is the Serret-Frenet parametrization.

The path P is characterized by a curvature $\kappa(s)$, which is the inversion of the radius of the circle tangent to the path at a point characterized by the parameter s. Let us consider a moving point M and the associated Serret-Frenet frame defined on the curve P by the normal and tangent unit vectors $\overrightarrow{n}$ and $\dfrac{d\overrightarrow{r}}{ds}$. The point M' is the orthogonal projection of the point M on the path P. It exists and is uniquely defined if the following conditions are satisfied, see e.g. [19]:

- the curvature $\kappa(s)$ is not bigger than $1/r_{\min} > 0$,
- the distance between the path P and the point M is smaller than $r_{\min}$,

where $r_{\min} = \dfrac{1}{\kappa_{\max}}$ and $\kappa_{\max} = \max_s \kappa(s)$. The coordinates of the point M relative to the Serret-Frenet frame are $(0, l)$ and relative to the basic frame X_0Y_0 are equal to (x, y), where l is the distance between M and M'. The curvilinear abscissa of M' is equal to s, where s is a distance along the path from some arbitrarily chosen point. The desired orientation of the platform satisfies the equation

$$\frac{\mathrm{d}\theta_r}{\mathrm{d}t} = \pm\kappa(s)\dot{s}. \tag{2}$$

The sign on the right side of the equation depends on the direction of moving along a desired curve (negative when the Serret-Frenet frame moves in the clockwise direction, positive otherwise).

If we want to express the position of the point M not in coordinates (x, y) relative to inertial frame, but relative to the given path P, we should use certain geometric relationships,

$$\dot{l} = (-\sin\theta_r \quad \cos\theta_r) \begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix}, \tag{3}$$

$$\dot{s} = \frac{(\cos\theta_r \quad \sin\theta_r)}{1 \mp \kappa(s)l} \begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix}, \tag{4}$$

where $\dot{x}$ and $\dot{y}$ can be expressed e.q. by the nonholonomic constraints of a mobile platform. In addition, we determine the orientation error

$$\widetilde{\theta} = \theta - \theta_r \tag{5}$$

and its derivative

$$\dot{\widetilde{\theta}} = \dot{\theta} - \dot{\theta}_r = \dot{\theta} \mp \kappa(s)\dot{s}. \tag{6}$$

The restrictions imposed on the desired path cause that inequality $|l\ \kappa(s)| < 1$ must hold.

For the unicycle the kinematic model expressed with respect to the Serret-Frenet frame is given by the following system of equations

$$\begin{cases} \dot{l} &= v\sin\widetilde{\theta}, \\ \dot{s} &= \dfrac{v\cos\widetilde{\theta}}{1 \mp \kappa(s)l}, \\ \dot{\widetilde{\theta}} &= \omega \mp \dfrac{\kappa(s)v\cos\widetilde{\theta}}{1 \mp \kappa(s)l}. \end{cases} \tag{7}$$

2.2. The chained form of a kinematic model. The kinematic equations of some of the mobile robots can be transformed into the chained form via a change of state and control variables, e.g. the equations of the unicycle and of the kinematic car. Such a transformation can be generalized to the kinematics models expressed with respect to the Serret-Frenet frame, which was shown in e.g. [20, 21].

The model described by Eq. (7) can be transformed into the three-dimensional chained system

$$\begin{aligned} \dot{z}_1 &= u_1, \\ \dot{z}_2 &= u_2, \\ \dot{z}_3 &= z_2 u_1 \end{aligned} \tag{8}$$

using the change of coordinates

$$\begin{aligned} z_1 &= s, \\ z_2 &= (1 \mp \kappa(s)l)\tan\widetilde{\theta}, \\ z_3 &= l \end{aligned} \tag{9}$$

and control variables

$$\begin{aligned} u_1 &= \frac{v\cos\widetilde{\theta}}{1 \mp \kappa(s)l}, \\ u_2 &= (\mp\dot{\kappa}(s)l \mp \kappa(s)u_1 z_2)\cdot\tan\widetilde{\theta} \\ &\quad + \frac{1 \mp \kappa(s)l}{\cos^2\widetilde{\theta}}(\omega \mp u_1\kappa(s))\,. \end{aligned} \tag{10}$$

It is worth to mention that the presented transformation is local $\left(|\widetilde{\theta}| < \dfrac{\pi}{2}\right)$.

3. Path following algorithms

There are different approaches to formulation of a path following task. The robot's motion can finish on stopping on a path or it may be continuous, cyclical. In this paper the latter approach is adopted. Let us assume that a direction of a movement along the desired curve is opposite to the clockwise direction, this means that

$$\frac{\mathrm{d}\theta_r}{\mathrm{d}t} = \kappa(s)\dot{s}.$$

Hence the equations describing parameters and path tracking errors have a below form

$$\begin{aligned} \dot{l} &= v\sin\widetilde{\theta}, \\ \dot{s} &= \frac{v\cos\widetilde{\theta}}{1 - \kappa(s)l}, \\ \dot{\widetilde{\theta}} &= \omega - \frac{\kappa(s)v\cos\widetilde{\theta}}{1 - \kappa(s)l}. \end{aligned} \tag{11}$$

3.1. Samson algorithm. For the path following errors for the mobile platform of (2,0) type expressed by the equations, see (11)

$$\dot{l} = v\sin\widetilde{\theta}, \tag{12}$$

$$\dot{\widetilde{\theta}} = u, \tag{13}$$

where $u = \omega - \dfrac{\kappa(s)\cos\widetilde{\theta}}{1 - \kappa(s)l}v$ is a new control for the second equation, Samson kinematic controller [7] is equal to

$$\begin{gathered} v = \text{const}, \qquad u = -k_2 l v\frac{\sin\widetilde{\theta}}{\widetilde{\theta}} - k_3\widetilde{\theta}, \\ k_2, k_3 > 0. \end{gathered} \tag{14}$$

3.2. Morin-Samson algorithm. The objective of Morin and Samson work was to design a control law which allows the vehicle to follow the path in a stable manner, independently of the sign of the longitudinal velocity. In [10] the authors proposed two kinematic control algorithms. The second one – which is a modification of their first proposal – is described here.

For the kinematic equations expressed with respect to a Serret-Frenet frame (11) which are transformed into the 3-dimensional chained system (8), the following control law is proposed

$$\begin{aligned} u_1 &= const, \\ u_2 &= -|u_1|k_0 z_0 - u_1 k_3 z_3 - |u_1|k_2 z_2 \\ &= -|u_1|k_0 \int_0^t u_1 z_3 - u_1 k_3 z_3 - |u_1|k_2 z_2, \end{aligned} \tag{15}$$

where a new variable z_0 is defined by

$$\dot{z}_0 = u_1 z_3, \qquad z_0(0) = 0.$$

If the below conditions are satisfied:

1. the polynomial $s^3 + k_3 s^2 + k_2 s + k_0$ is Hurwitz stable (exceptionally in this line the notation s means the parameter of the Laplace transform),
2. the initial conditions verify

$$z_3^2(0) + \frac{1}{k_3 - \dfrac{k_0}{k_2}} z_2^2(0) < \frac{1}{\kappa_{\max}^2}$$

then the constraint $|l\ \kappa(s)| < 1$ is satisfied along any solution to the controlled system.

4. Probabilistic approach to robot control

4.1. Motivation. The often practice in a robotic control systems design is a deterministic approach. But in real-life applications, there are usually met some uncertainties [12], as distortions of used devices or inaccuracy of sensors. The motivation for consideration of a noised control signal is its likeness to real robotic systems. The probabilistic approach [14] places the robot in the physical world, as opposed to a deterministic approach with ideal conditions modelled in computer simulations. Making real of the world in which we place the robot should deal with mentioned uncertainties (due to the used sensors accuracy) and changing external conditions (like e.g. an earthquake). The first type of internal interference can be estimated, based on the sensor unit accuracy specified by a manufacturer. The external interference can be extremely difficult to predict and to assess its exact impact on the robot.

Taking into account those limitations, in this article we will only concern the measurement inaccuracy of devices, and more – only the influence of noise (or measurement uncertainty) to determine the orientation of the robot. In the analysed kinematic path following algorithms, Morin-Samson and Samson, the parameter that has the biggest influence on the control is the orientation θ.

4.2. Noised orientation. We accept the convention that random variables and stochastic processes will be marked in capital letters, with the exception of ϵ – noise distorting the orientation. The orientation θ was noised at every step of feedback loop between an object and a controller. The loop steps were chosen by stochastic differential equations solver 'ode45' in a numerical computing environment MATLAB. For this reason, the theoretically continuous in time orientation θ, the noise ϵ and other random variables and stochastic processes are discrete in time in performed simulations. So in the following sections the time t means discrete moments in time, defined by SDE solver.

There were three variables describing the kinematics of a robot: an orientation θ in radians and a robot location coordinates $(x,\ y)$. The only variable noised before transmission to the controller was the orientation θ. For a fixed time t, the orientation $\theta(t)$ was distorted by a *Gaussian white noise* with a variance σ^2, denoted by $\sigma\epsilon(t)$. Generally, the white noise process $\epsilon(t)$ has the following properties:

- is independent and identically distributed (*iid*),
- has zero mean function, i.e.

$$\mathbb{E}\{\epsilon(t)\} = 0 \quad \text{for} \quad t > 0, \tag{16}$$

- is uncorrelated, i.e. its autocorrelation function

$$\mathbb{E}\{\epsilon(t_1)\epsilon(t_2)\} = \delta(t_1 - t_2) \quad \text{for} \quad t_1,\ t_2 > 0, \tag{17}$$

where $\delta(\tau)$ is Kronecker delta, $\delta(\tau) = \begin{cases} 1, & \text{if } \tau = 0 \\ 0, & \text{if } \tau \neq 0 \end{cases}$.

We assume that $\epsilon(t)$ is normally distributed $\epsilon \sim \mathcal{N}(0,\ 1)$, for a fixed time t, because Gaussian noise is a distortion the most commonly appearing in nature. The noised orientation θ is denoted by Θ_ϵ:

$$\Theta_\epsilon = \theta + \sigma\epsilon \quad \text{for all} \quad t, \tag{18}$$

where σ has a fixed value, but usually unknown and only estimated.

4.3. Theoretical properties of errors. The influence of an additive Gaussian white noise on analysed algorithms were examined taking into account:

- a random orientation error $\widetilde{\Theta}$ (a random equivalent of a deterministic orientation error $\widetilde{\theta}$),
- a random distance error L (a random equivalent of a deterministic distance error l).

Corollary 4.1. The orientation error $\widetilde{\Theta}$ for a distorted orientation Θ_ϵ is a random process and has a distribution given by the mean function $\mu_{\widetilde{\Theta}}$ and the autocorrelation function $\gamma_{\widetilde{\Theta}}$:

$$\mu_{\widetilde{\Theta}}(t) = \mathbb{E}\{\widetilde{\Theta}(t)\} = \theta(t) - \theta_r(t) \quad \text{for} \quad t > 0, \tag{19}$$

$$\begin{aligned} \gamma_{\widetilde{\Theta}}(t_1, t_2) &= \mathbb{E}\{\widetilde{\Theta}(t_1)\widetilde{\Theta}(t_2)\} \\ &= [\theta(t_1) - \theta_r(t_1)][\theta(t_2) - \theta_r(t_2)] + \sigma^2\delta(t_1 - t_2) \\ &\text{for} \quad t_1,\ t_2 > 0. \end{aligned} \tag{20}$$

Proof. Firstly, the random process Θ_ϵ is plugged in the place of a deterministic orientation θ for all t in the formula (5)

$$\widetilde{\theta} = \theta - \theta_r.$$

For a distorted orientation Θ_ϵ, we get that the orientation error $\widetilde{\Theta}$ is equal to:

$$\widetilde{\Theta} = \Theta_\epsilon - \theta_r = (\theta + \sigma\epsilon) - \theta_r = (\theta - \theta_r) + \sigma\epsilon, \tag{21}$$

where $(\theta - \theta_r)$ is deterministic function in time. The distribution of a white noise ϵ is given by Eqs. (16) and (17). The noise level σ has a fixed value. The mean function is equal to

$$\mu_{\widetilde{\Theta}}(t) = \mathbb{E}\{\widetilde{\Theta}(t)\} = \theta(t) - \theta_r(t) + \sigma\mathbb{E}\{\epsilon(t)\}, \tag{22}$$

while the autocorrelation function is equal to

$$\begin{aligned}\gamma_{\widetilde{\Theta}}(t_1, t_2) &= \mathbb{E}\{\widetilde{\Theta}(t_1)\widetilde{\Theta}(t_2)\} \\ &= \mathbb{E}\{[\theta(t_1) - \theta_r(t_1) + \sigma\epsilon(t_1)] \\ &\quad \cdot[\theta(t_2) - \theta_r(t_2) + \sigma\epsilon(t_2)]\} \\ &= [\theta(t_1) - \theta_r(t_1)][\theta(t_2) - \theta_r(t_2)] \\ &\quad + [\theta(t_1) - \theta_r(t_1)]\sigma\mathbb{E}\{\epsilon(t_2)\} \\ &\quad + [\theta(t_2) - \theta_r(t_2)]\sigma\mathbb{E}\{\epsilon(t_1)\} + \sigma^2\mathbb{E}\{\epsilon(t_1)\epsilon(t_2)\}.\end{aligned} \tag{23}$$

Corollary 4.2. Under assumption that $\theta(t) = \theta_r(t)$ for $t > T$, the orientation error $\widetilde{\Theta}$ for a distorted orientation Θ_ϵ is a Gaussian white noise with a stationary Gaussian distribution with a zero mean and a variance σ^2, i.e.

$$\widetilde{\Theta} \sim \mathcal{N}(0,\, \sigma^2) \quad \text{for} \quad t > T. \tag{24}$$

Proof. Using the Corollary 4.3 with the assumption $\theta(t) - \theta_r(t) = 0$, we have

$$\mu_{\widetilde{\Theta}}(t) = \mathbb{E}\{\widetilde{\Theta}(t)\} = 0 \quad \text{for} \quad t > T, \tag{25}$$

$$\begin{aligned}\gamma_{\widetilde{\Theta}}(t_1, t_2) &= \mathbb{E}\{\widetilde{\Theta}(t_1)\widetilde{\Theta}(t_2)\} \\ &= \sigma^2\delta(t_1 - t_2) = \begin{cases} \sigma^2, & \text{if} \quad t_1 = t_2 \\ 0, & \text{if} \quad t_1 \neq t_2 \end{cases} \\ &\text{for} \quad t_1,\, t_2 > T.\end{aligned} \tag{26}$$

Thus, the orientation error $\widetilde{\Theta}$ is uncorrelated and has a zero mean. The normal distribution is a consequence of the equation $\widetilde{\Theta} = \sigma\epsilon$.

Corollary 4.3. For Morin-Samson algorithm when the Serret-Frenet parametrization is correct, the derivative of the distance error $\dot{L}$ for a distorted orientation Θ_ϵ satisfies the diferential equation

$$\dot{L} = u_1 \tan\widetilde{\Theta}(1 \mp \kappa(s)L), \tag{27}$$

where $u_1 = \text{const}$ and $\kappa(s) = \kappa(s(t))$.

Proof. The derivative of the deterministic distance error $\dot{l}$ satisfies the equation

$$\dot{l} = v\sin\widetilde{\theta} = u_1 \tan\widetilde{\theta}(1 \mp \kappa(s)l). \tag{28}$$

In the place of the deterministic orientation error $\widetilde{\theta}$, there is the random variable $\widetilde{\Theta}$ plugged-in. The longitudinal velocity is given by the equation

$$v = \frac{u_1(1 \mp \kappa(s)l)}{\cos\widetilde{\theta}}, \tag{29}$$

where $|\widetilde{\theta}| < \dfrac{\pi}{2}$ and $u_1 = \text{const}$, see the Morin-Samson kinematic controller in (15).

Corollary 4.4. For Samson algorithm when the Serret-Frenet parametrization is correct, under assumption that $\theta(t) \approx \theta_r(t)$ for $t > T$, the derivative of the distance error $\dot{L}$ for a distorted orientation Θ_ϵ can be approximated for a small values of $\sigma\epsilon$ $\left(\text{i.e. } |\sigma\epsilon| < \dfrac{\pi}{8}\right)$ by a Gaussian white noise $v\sigma\epsilon$ with a stationary Gaussian distribution with a zero mean and a variance $(v\sigma)^2$, i.e.

$$\dot{L} \approx v\sigma\epsilon \sim \mathcal{N}(0,\, (v\sigma)^2) \quad \text{for} \quad t > T, \tag{30}$$

where v is a longitudinal velocity and it has a constant value, and σ is a noise level of the orientation Θ_ϵ.

Proof. Plugging-in the random variable $\widetilde{\Theta}$ in the place of a deterministic orientation error $\widetilde{\theta}$ in the formula (7)

$$\dot{l} = v\sin(\widetilde{\theta}),$$

and considering the fact that $v = \text{const}$ (see Eq. (14)), the derivative of the distance error $\dot{L}$ for a distorted orientation Θ_ϵ has the form:

$$\begin{aligned}\dot{L} &= v\sin(\widetilde{\Theta}) = v\sin(\Theta_\epsilon - \theta_r) \\ &= v\sin((\theta - \theta_r) + \sigma\epsilon) \\ &= v(\sin(\theta - \theta_r)\cos(\sigma\epsilon) + \cos(\theta - \theta_r)\sin(\sigma\epsilon)).\end{aligned} \tag{31}$$

For $\theta - \theta_r \approx 0$, using the properties of sine and cosine functions we have

$$\dot{L} = v\sin(\sigma\epsilon).$$

Sine function can be approximated with a linear function for small angles, i.e. for $\sigma\epsilon \in \left(-\dfrac{\pi}{8},\, \dfrac{\pi}{8}\right)$. Therefore,

$$\dot{L} \approx v\sigma\epsilon \sim \mathcal{N}(0,\, (v\sigma)^2).$$

5. Simulations and tests

The task of the robot was to achieve a defined path – a circle with center in the point $(0,\, 0)$ and a radius equal to 2 – and to continue its movement along the path. The controller's parameters were constant, $v = 1, k_2, k_3 = 1$ for Samson algorithm and $u_1 = 1, k_0 = 1, k_2, k_3 = 1$ for Morin-Samson algorithm. The additive noise $\sigma\epsilon(t)$ was generated for several levels of dispersion σ: 10^{-4}, 10^{-3}, 10^{-2}, 0.1, 0.2, …, 0.9, and 1. For the noised orientation $\widehat{\Theta}_\epsilon$, the two following errors were investigated:

- an orientation error $\widetilde{\Theta}$ (in radians),
- a distance error L (in meters).

The simulations were performed for two algorithms: Morin-Samson and Samson. The robot had 25 seconds to get and follow the defined path. The robot starts to move inside the circle, which constitutes the path the robot should follow. The simulation results for chosen values of σ were presented in Figs. 2 and 3. For orientation and distance errors, the sample mean values ($\widehat{\mu}_{\widetilde{\Theta}}$ and $\widehat{\mu}_L$) and the sample standard deviation ($\widehat{\sigma}_{\widetilde{\Theta}}$ and $\widehat{\sigma}_L$) were calculated for the last few seconds of simulations and placed in Tables 2 and 3. There was also presented $3\widehat{\sigma}$-interval around the mean value $\widehat{\mu}$. For the Gaussian distribution, 99.7% of observations can be found inside the interval $[\widehat{\mu} - 3\widehat{\sigma},\, \widehat{\mu} + 3\widehat{\sigma}]$.

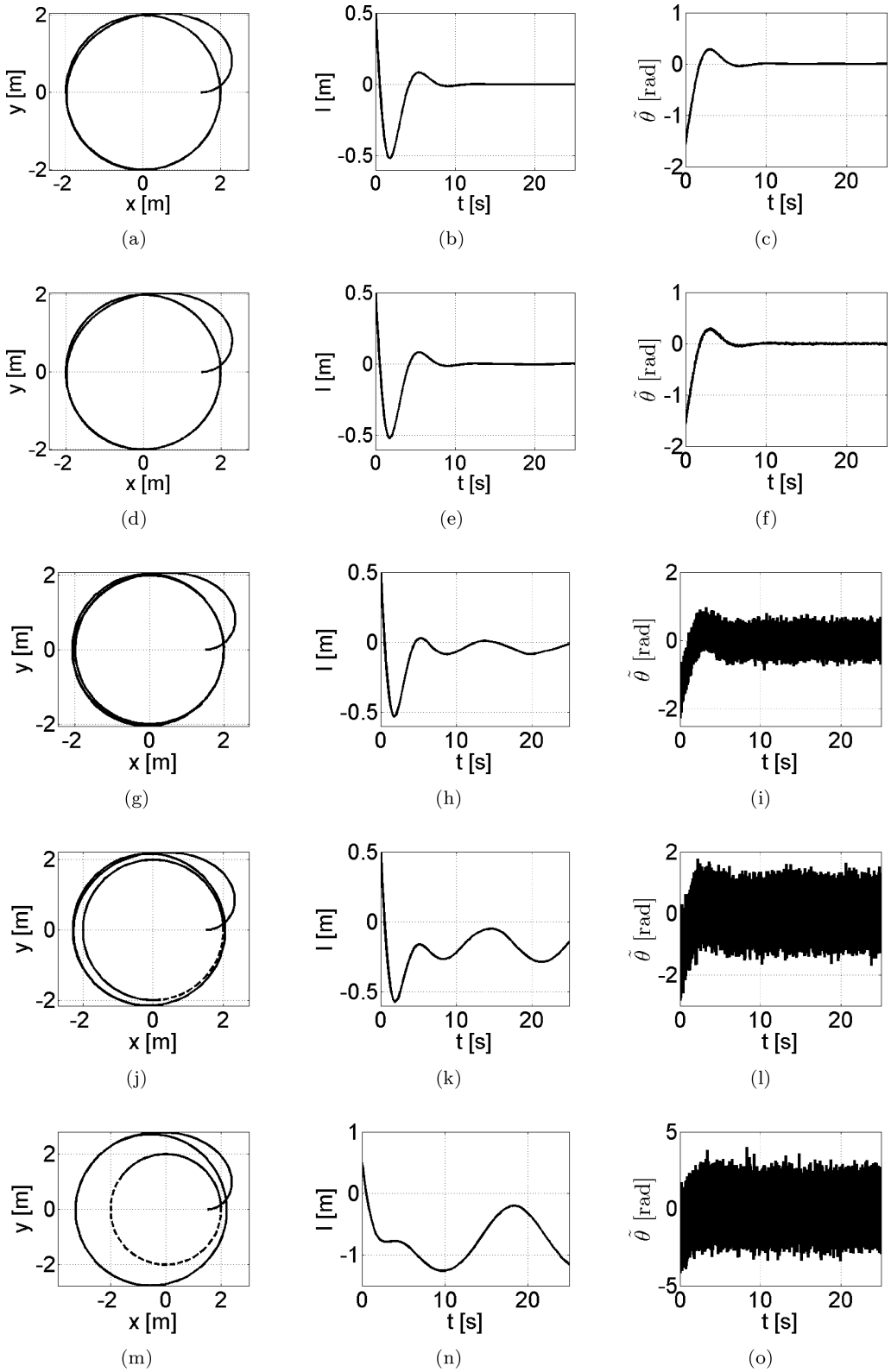

Fig. 2. The path following (control algorithm – Samson): (a), (d), (g), (j), (m) – XY plane, (b), (e), (h), (k), (n) – the distance error l, (c), (f), (i), (l), (o) – the orientation error $\tilde{\theta}$, $\sigma = 0$ for (a)–(c), $\sigma = 0.1$ for (d)–(f) $\sigma = 0.2$ for (g)–(i), $\sigma = 0.4$ for (j)–(l), $\sigma = 0.8$ for (m)–(o)

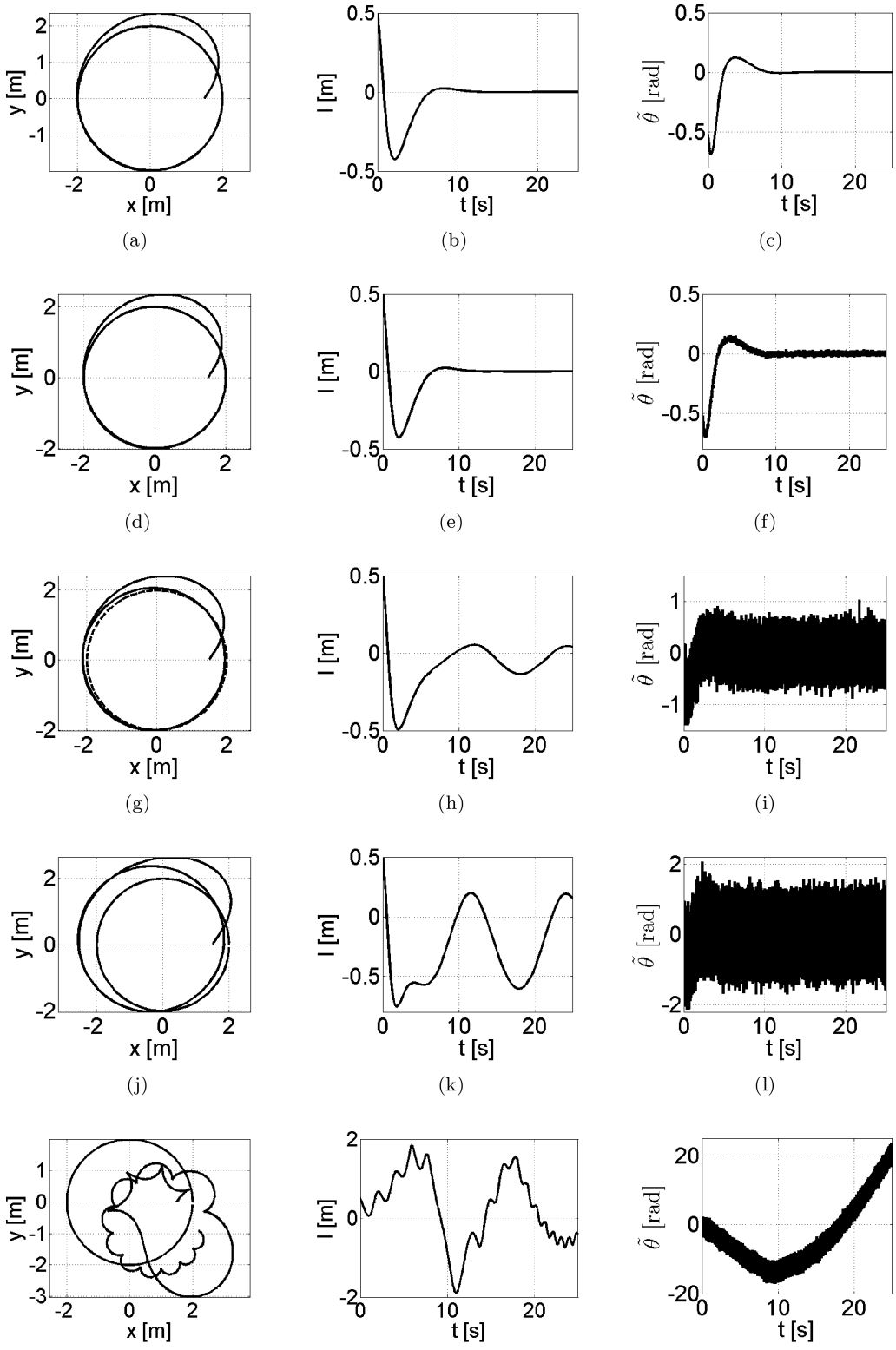

Fig. 3. The path following (control algorithm – Morin-Samson): (a), (d), (g), (j), (m) – XY plane, (b), (e), (h), (k), (n) – the distance error l, (c), (f), (i), (l), (o) – the orientation error $\widetilde{\theta}$, $\sigma = 0$ for (a)–(c), $\sigma = 0.1$ for (d)–(f), $\sigma = 0.2$ for (g)–(i), $\sigma = 0.4$ for (j)–(l), $\sigma = 0.8$ for (m)–(o)

Table 1
The results for Samson algorithm (the orientation error $\widetilde{\Theta}$)

σ	S	N	Z	$\widehat{\mu}_{\widetilde{\Theta}}$	$\widehat{\sigma}_{\widetilde{\Theta}}$	$[\widehat{\mu}_{\widetilde{\Theta}} - 3\widehat{\sigma}_{\widetilde{\Theta}}, \quad \widehat{\mu}_{\widetilde{\Theta}} + 3\widehat{\sigma}_{\widetilde{\Theta}}]$
0.0001	yes	yes	yes	0.00001	0.00011	[-0.00031, 0.00034]
0.0010	yes	yes	yes	−0.00006	0.00112	[-0.00343, 0.00330]
0.0100	yes	yes	yes	−0.00028	0.01006	[-0.03046, 0.02990]
0.1000	yes	yes	yes	−0.00004	0.10138	[-0.30417, 0.30409]
0.2000	yes	yes	yes	−0.00078	0.19956	[-0.59946, 0.59790]
0.3000	yes	yes	yes	0.00292	0.29947	[-0.89550, 0.90134]
0.4000	yes	yes	yes	0.00675	0.39805	[-1.18739, 1.20089]
0.5000	yes	yes	no	0.01493	0.50039	[-1.48625, 1.51611]
0.6000	yes	yes	yes	0.00832	0.59219	[-1.76825, 1.78489]
0.7000	yes	yes	yes	0.00888	0.70940	[-2.11933, 2.13708]
0.8000	yes	yes	yes	0.00980	0.80530	[-2.40609, 2.42569]
0.9000	yes	yes	yes	−0.01133	0.89635	[-2.70039, 2.67773]
1.0000	yes	no	yes	−0.00856	1.00053	[-3.01014, 2.99303]

Table 2
The results for Morin-Samson algorithm (the orientation error $\widetilde{\Theta}$)

σ	S	N	Z	$\widehat{\mu}_{\widetilde{\Theta}}$	$\widehat{\sigma}_{\widetilde{\Theta}}$	$[\widehat{\mu}_{\widetilde{\Theta}} - 3\widehat{\sigma}_{\widetilde{\Theta}}, \quad \widehat{\mu}_{\widetilde{\Theta}} + 3\widehat{\sigma}_{\widetilde{\Theta}}]$
0.0001	no	yes	yes	0.00002	0.00014	[−0.00040, 0.00044]
0.0010	yes	yes	yes	−0.00009	0.00109	[−0.00336, 0.00317]
0.0100	yes	yes	yes	0.00037	0.00992	[−0.02938, 0.03013]
0.1000	yes	yes	yes	0.00032	0.09956	[−0.29837, 0.29902]
0.2000	yes	yes	yes	−0.00230	0.20046	[−0.60370, 0.59909]
0.3000	yes	yes	no	−0.00650	0.29726	[−0.89829, 0.88528]
0.4000	yes	yes	yes	0.00191	0.40019	[−1.19866, 1.20248]
0.5000	yes	yes	yes	0.00306	0.49965	[−1.49589, 1.50201]
0.6000	yes	yes	no	0.01949	0.60743	[−1.80280, 1.84179]
0.7000	yes	yes	yes	−0.00786	0.69690	[−2.09856, 2.08284]
0.8000	yes	no	no	20.68008	0.80196	[18.27419, 23.08597]
0.9000	yes	yes	no	−82.45772	0.91320	[−85.19731, −79.71813]
1.0000	yes	yes	no	−80.61119	1.02498	[−83.68611, −77.53626]

Table 3
The results for Samson algorithm (the distance error L)

σ	S	$\widehat{\mu}_L$	$\widehat{\sigma}_L$	$[\widehat{\mu}_L - 3\widehat{\sigma}_L, \quad \widehat{\mu}_L + 3\widehat{\sigma}_L]$
0.0001	no	0.00007	0.00012	[−0.00030, 0.00045]
0.0010	no	0.00035	0.00091	[−0.00238, 0.00307]
0.0100	no	−0.00092	0.00248	[−0.00835, 0.00651]
0.1000	no	−0.01562	0.00536	[−0.03172, 0.00047]
0.2000	no	−0.04447	0.01563	[−0.09136, 0.00241]
0.3000	no	−0.08786	0.02234	[−0.15489, −0.02083]
0.4000	no	−0.19513	0.02857	[−0.28084, −0.10941]
0.5000	no	−0.36274	0.02385	[−0.43428, −0.29120]
0. 6000	no	−0.61677	0.01323	[−0.65646, −0.57707]
0.7000	no	−0.92244	0.00609	[−0.94072, −0.90416]
0.8000	no	−1.10745	0.02807	[−1.19166, −1.02324]
0.9000	no	−1.09119	0.04750	[−1.23369, −0.94869]
1.0000	no	−0.78942	0.05382	[−0.95086, −0.62797]

Table 4
The results for Morin-Samson algorithm (the distance error L)

σ	S	$\widehat{\mu}_L$	$\widehat{\sigma}_L$	$[\widehat{\mu}_L - 3\widehat{\sigma}_L, \quad \widehat{\mu}_L + 3\widehat{\sigma}_L]$
0.0001	no	−0.00000	0.00002	[−0.00006, 0.00005]
0.0010	no	0.00021	0.00023	[−0.00048, 0.00089]
0.0100	no	−0.00025	0.00105	[−0.00339, 0.00289]
0.1000	no	0.01113	0.00292	[0.00236, 0.01990]
0.2000	no	0.04370	0.00160	[0.03889, 0.04851]
0.3000	no	0.13790	0.00573	[0.12070, 0.15509]
0.4000	no	0.17544	0.01195	[0.13958, 0.21130]
0.5000	no	0.13543	0.00584	[0.11789, 0.15296]
0.6000	no	−0.08654	0.01818	[−0.14107, −0.03201]
0.7000	no	−2.37433	0.00759	[−2.39709, −2.35156]
0.8000	no	−0.39407	0.00797	[−0.41798, −0.37017]
0.9000	no	0.18410	0.02158	[0.11937, 0.24884]
1.0000	no	0.28838	0.01465	[0.24443, 0.33233]

The test data come from the final seconds of the simulation. The number of SDE solver steps in the 25-second period of time is variable for different noise levels and the greater the higher the noise level: 58 steps for $\sigma = 0.0001$, 391 – 0.001, 3572 – 0.01, 35305 – 0.1, 71678 – 0.2, ..., 395427 – 1.0. The final time periods were chosen arbitrarily, eg. 7 sec (only 10 steps) for $\sigma = 0.0001$, 3.5 sec (10k steps) for $\sigma = 0.2$ and 1.7 sec (10k steps) for $\sigma = 1.0$. The time periods for analysis were

selected in such a way that the robot has a chance to reach the desired path and stabilize its movement. In some cases the desired path was not reached. Additionally, the orientation error and distance from path were not stabilized around zero, but for high noise levels were still increasing in final seconds of the simulation.

The data from the end of simulation were tested for distribution stationarity by the Kwiatkowski, Phillips, Schmidt, and Shin test [22] (in abbreviation KPSS test), see Subsec. 5.1. For cases with stationary distributions, there were also performed tests of distribution normality by the Lilliefors test, presented in Subsec. 5.2. Zero mean value was tested by Student t-test, described in Subsec. 5.3. All the tests were performed at the significance level 5%. For data in Tables 1–4, we introduce the following notation for columns:

S – the distribution stationarity for $\widetilde{\Theta}$ or L (the KPSS test [22]),
N – the distribution normality for $\widetilde{\Theta}$ (the Lilliefors test [23]),
Z – zero mean test for $\widetilde{\Theta}$ (the Student's t-test [24]).

The normality and zero mean tests were conducted only for the orientation error $\widetilde{\Theta}$, since for the distance error L the null hypothesis of distribution stationarity was always rejected. And the stationarity of distribution is a primary assumption required in the both mentioned tests.

5.1. Stationarity test. The definition of *the weak-sense stationarity* says that 1st and 2nd moments of a random processes $x(t)$ are needed to not vary with respect to time. It means that a continuous-time random process $x(t)$ is stationary, if its mean function fulfils the equation

$$\mathbb{E}\{x(t)\} = \mu_x(t) = \mu_x(t+\tau) \quad \text{for all} \quad \tau \in \mathbb{R} \tag{32}$$

and autocorrelation function is equal to

$$\begin{gathered}\mathbb{E}\{x(t_1)x(t_2)\} = \gamma_x(t_1,t_2) \\ = \gamma_x(t_1+\tau,t_2+\tau) = \gamma_x(t_1-t_2,0) \\ \text{for all} \quad \tau \in \mathbb{R}.\end{gathered} \tag{33}$$

The Kwiatkowski, Phillips, Schmidt, and Shin (KPSS) test [22] verifies the null hypothesis that a univariate time series $x(t)$ is stationary against the alternative hypothesis that it is a nonstationary unit-root process:

$$\begin{aligned}&\text{H0: } x(t) = c + dt + \xi(t) \\ &\qquad \text{(trend-stationary process),} \\ &\text{H1: } \triangle x_{t-1,t} = x(t) - x(t-1) = d + \xi(t) \\ &\qquad \text{(a difference-stationary process).}\end{aligned} \tag{34}$$

Although the KPSS test allow to check whether the time series $x(t)$ is stationary with a deterministic trend, we used the test for the null hypothesis without trend, i.e. with level trend $d = 0$.

5.2. Normality test. To verify the hypothesis of a normal distribution we use nonparametric Lilliefors test [23], also known as the Kolmogorov-Smirnov test with Lilliefors correction. The Lilliefors test requires an assumption that analysed sample $\{x_i\}_{i=1}^n$ is iid, in particular it means that the distribution is stationary. The test statistics is:

$$D_n = \max_{x_i} \left| \Phi\left(\frac{x_i - \overline{x}}{S}\right) - F_n\left(x_i\right) \right|, \tag{35}$$

where Φ – the cumulative standard normal $\mathcal{N}(0,\,1)$ distribution function, F_n – empirical cumulative distribution function from n-element sample, $\overline{x} = \dfrac{1}{n}\sum\limits_{i=1}^{n} x_i$ – sample mean, $S^2 = \dfrac{1}{n-1}\sum\limits_{i=1}^{n}(x_i - \overline{x})^2$ – sample variance.

5.3. Test of a zero mean. The one sample Student's t-test [24] was used for the examination of the null hypothesis that the mean is equal to zero:

$$\begin{aligned}&\text{H0: } \mu_x = 0, \\ &\text{H1: } \mu_x \neq 0.\end{aligned} \tag{36}$$

The test assumes that the sample is normally distributed. The sample $\{x_i\}_{i=1}^n$ form measured observations. The rejection area is chosen to be two-tailed, because there are no predictions if values of μ_x are positive or negative. The test statistics for n-element sample has the form:

$$t = \frac{\overline{x}}{S}\sqrt{n}, \tag{37}$$

where the symbols $\overline{x}$ and S are the same as in Subsec. 5.2, sample mean and sample standard deviation, respectively.

5.4. Simulation results. Orientation error. In both algorithms Morin-Samson and Samson, the values of sample deviation $\widehat{\sigma}_{\widetilde{\Theta}}$ are close to the noise level σ, i.e. $\widehat{\sigma}_{\widetilde{\Theta}} \approx \sigma$. The orientation error is transferred directly without multiplication and it does not accumulate. The noise level of the orientation Θ_ϵ (see (18)) was transferred on the noise level of the orientation $\widetilde{\Theta}$.

According to Corollary 4.2, for $\widehat{\mu}_{\widetilde{\Theta}} \approx 0$ (i.e. $\theta - \theta_r \approx 0$) $\widetilde{\Theta} \sim \mathcal{N}(0,\,\sigma^2)$. It can be observed for all noise levels in Samson algorithm, and for $\sigma = 10^{-4}, \ldots, 0.7$ in Morin-Samson algorithm. In such a case one can use 3σ rule – 99.7% of samples lie within the range $(-3\sigma,\,3\sigma)$, what was confirmed by simulations, see Tables 1 and 2. The Student's t-test used for the examination of the null hypothesis that the mean is equal to zero, did not reject the null hypothesis for Samson algorithm, so we could assume that the sample mean is equal to zero. Similar results were obtained for Morin-Samson algorithm when $\sigma \leq 0.7$. For $\sigma > 0.7$ one cannot use Corollary 4.2 as the assumption that $\theta - \theta_r \approx 0$ does not hold anymore.

For high values of σ Samson algorithm deals better with the task than the Morin-Samson algoritm. The Samson algorithm retained the shape of the desired path, while the path obtained by using the Morin-Samson algorithm does not even resemble the ellipse.

Distance error. For the Samson algorithm for noise levels $\sigma \leq 0.2$, the mean distance error (in meters) $\widehat{\mu}_L < 0.05$. It means that the error was lower than 5 cm. For noise levels $\sigma > 0.2$ the mean distance error $\widehat{\mu}_L$ is unacceptably high, even higher than 110 cm. One has to take into account that while the noise level increases, the risk that the robot moves too far from the path increases too and the applied Serret-Frenet parametrization could be no longer correct, not to mentioned the fact that the distance error L is influenced by the noise appearing in $\widetilde{\Theta}$. This is probably why for $\sigma > 0.1$ the path circled by the robot is not the same as the desired path – it is shifted and scaled. The higher the noise level was, the bigger deformations of the followed path comparing to the desired path were observed.

For the Morin-Samson algorithm for the noise level $\sigma > 0.2$, the mean distance error $\widehat{\mu}_L$ is unacceptably high. The error has even a value of 237 cm. For the noise level $\sigma \leq 0.2$ the mean $\widehat{\mu}_L$ was close to zero.

For both algorithms, Morin-Samson and Samson, all tests of stationarity (column S in Tables 3 and 4) rejected the null hypothesis that the distribution of L is stationary. The stationarity would mean that the distribution of L does not change in time, i.e. its mean and the autocorrelation functions are constant in time. The KPSS test rejected the hypothesis about a stationarity in all cases. Thus either the expected value or the autocorrelation function depend on the time. Due to the fact that the distribution is not stationary, there is no point in running normality and zero mean tests because those test could be run only for a random sample, i.e. a realization of a random variable with a stationary distribution. For $\sigma \leq 0.2$ the mean was close to zero and quite huge for high values of σ.

Taking into account the simulation results one may conclude that the Samson algorithm is more noise robust than Morin-Samson algorithm. Before Morin-Samson algorithm could be used, two nonlinear transformations of the robot have to be computed, while for Samson algorithm one transformation is enough. In addition both the Serret-Frenet parametrization and the transformation to the chained form are local. This means that during the control process not only the curvature of the desired path is constrained and the distance between the robot and the path must be limited, but also $\widetilde{\theta}$ must be bounded: $|\widetilde{\theta}| < \dfrac{\pi}{2}$, and then the expression $\tan\widetilde{\theta}$ is finite: $|\tan\widetilde{\theta}| < \infty$, see (9). The errors made at every step of the feedback control loop resulting from inaccuracy of measurement devices can accumulate and an additional nonlinear transformation may cause that the influence of a noise on a task realization increases. For low noise levels both presented algorithms work properly, while for bigger values of σ Samson algorithm works better than Morin-Samson algorithm.

6. Conclusions

The presented paper can be seen as a report illustrating the particular properties of the closed-loop system such a robustness to external disturbances.

In general it is recommended to gather information about the location of the robot as often as possible and measure the robot's orientation precisely (the error equal to few degrees is acceptable). The frequent regular feedback allows to correct the robot's behaviour more often and helps to deal better with uncertainty. When obtaining information from measurement devices several or more (whenever possible) measurements should be done. Then the measurements could be averaged. Assuming that the noise has a zero mean, one can conclude that according to the Law of Large Numbers the averaged measurement result converges to the expected value.

The problem related to examining a noise sensitivity of control algorithms for robotic objects is quite important and is still not well considered in the robotics literature. The main contribution of this work is to show that it is possible to verify the effectiveness of some of the control algorithms in the presence of the noise in the control loop based on simulation results and statistical techniques.

In the presented paper it was assumed that the noise is centered, i.e. the measurement devices are correctly calibrated. The situation might be more complex when the devices would not be calibrated precisely. Then the noise with a non zero mean is transferred. Further research should be carried out to investigate what is the required calibration accuracy of measurement devices for the algorithms to work properly. The research can be extended by taking into consideration not only kinematics, but also dynamics of the robot.

Acknowledgements. This work was supported by the fellowship co-financed by the European Union within the European Social Fund.

REFERENCES

[1] Z. Li and J.F. Canny, *Nonholonomic Motion Planning*, vol. 192, Springer, Berlin, 1993.

[2] A.M. Bloch, *Nonholonomic Mechanics and Control*, vol. 24, Springer, Berlin, 2003.

[3] I. Dulęba, "Impact of control representations on efficiency of local nonholonomic motion planning", *Bull. Pol. Ac.: Tech.* 59 (2), 213–218 (2011).

[4] G. Campion and W. Chung, "Wheeled robots", in *Handbook of Robotics*, eds. B. Siciliano and O. Khatib, Springer-Verlag, Berlin, 2008.

[5] J.B. Pomet, "Explicit design of time-varying stabilizing control laws for a class of controllable systems without drift", *Systems & Control Letters* 18, 147–158 (1992).

[6] A. Mazur, "Hybrid adaptive control laws solving a path following problem for nonholonomic mobile manipulators", *Int. J. Control* 77 (15), 1297–1306 (2004).

[7] C. Samson, "Path following and time-varying feedback stabilization of a wheeled mobile robots", *Proc. IEEE Int. Conf. on Advanced Robotics and Computer Vision* 1, 1.1–1.5 (1992).

[8] D. Soetanto, L. Lapierre, and A. Pascoal, "Adaptive, non-singular path-following control of dynamic wheeled robots", *Proc. IEEE Conf. on Decision and Control* 2, 1765–1770 (2003).

[9] A. Morro, A. Sgorbissa, and R. Zaccaria, "Path following for unicycle robots with an arbitrary path curvature", *IEEE Trans. on Robotics* 27 (5), 1016–1023 (2011).

[10] P. Morin and C. Samson, "Motion control of wheeled mobile robots", in *Handbook of Robotics*, eds. B. Siciliano and O. Khatib, Springer-Verlag, Berlin, 2008.

[11] A. De Luca, G. Oriolo, and C. Samson, "Feedback control of a nonholonomic car-like robot", in *Robot Motion Planning and Control* ed. J.-P. Laumond, pp. 171–253, Springer-Verlag, Berlin, 1998.

[12] L. Meeden, *Bridging the Gap Between Robot Simulations and Reality with Improved Models of Sensor Noise*, Morgan Kaufman Publishers, San Francisco, 1998.

[13] O. Miglino, H.H. Lund, and S. Nolfi, "Evolving mobile robots in simulated and real environments", *Artificial Life* 2, 417–434 (1996).

[14] S. Thrun, W. Burgard, and D. Fox, "Probabilistic robotics", in *Intelligent Robotics and Autonomous Agents*, MIT Press, London, 2005. ISBN 9780262201629, LCCN 2005043346.

[15] D. A. Lizárraga, P. Morin, and C. Samson, "Non-robustness of continuous homogeneous stabilizers for affine control systems", *Proc. 38th IEEE Conf. on Decision and Control* 1, 855–860 (1999).

[16] Ch. Prieur and A. Astolfi, "Robust stabilization of chained systems via hybrid control", *IEEE Trans. on Automatic Control* 48 (10), 1768–1772 (2003).

[17] G. Oriolo, A. De Luca, and M. Vendittelli, "WMR control via dynamic feedback linearization: design, implementation, and experimental validation", *IEEE Trans. on Control Systems Technology* 10 (6), 835–852 (2002).

[18] D. Pazderski, B. Krysiak, and K. Kozłowski, "A comparison study of discontinuous control algorithms for a three-link nonholonomic manipulator", in *Robot Motion and Control 2011*, pp. 377–389, Springer, Berlin, 2012.

[19] A. Fradkov, I. Miroshnik, and V. Nikiforov, *Nonlinear and Adaptive Control of Complex Systems*, Kluwer Academic Publishers, Dordrecht, 1999.

[20] J. Płaskonka, "The path following control of a nonholonomic car-like robot", *ICT Young 2012 – Conference Materials* 1, 573–579 (2012).

[21] J. Płaskonka, "The path following control of a unicycle based on the chained form of a kinematic model derived with respect to the Serret-Frenet frame", *17th Int. Conf. on Methods and Models in Automation & Robotics* 1, CD-ROM (2012).

[22] D. Kwiatkowski, P.C.B. Phillips, P. Schmidt, and Y. Shin, "Testing the null hypothesis of stationarity against the alternative of a unit root: How sure are we that economic time series have a unit root?" *J. Econometrics* 54 (1–3), 159–178 (1992).

[23] H. Lilliefors, "On the Kolmogorov-Smirnov test for normality with mean and variance unknown", *J. American Statistical Association* 62, 399–402 (1967).

[24] D.C. Montgomery and G.C. Runger, *Applied Statistics and Probability for Engineers*, John Wiley & Sons, London, 2010.