Abstract—The main contribution of this work is associated to the fault detection and diagnosis (FDD) of sensors & actuators, and fault accommodation applied to wheeled mobile robots. FDD (i.e. fault detection, isolation, and estimation) is done by generating and analyzing residuals with different signatures. Estimated faults are then compensated to be tolerated. Simulation results show that this method is very promising.

Keywords: Fault diagnosis (FDD), Unicycle mobile robot, Fault accommodation.

I. INTRODUCTION

Thanks to their ability to easily perform hard, risky, repetitive, and expensive activities, wheeled mobile robots (WMRs) are increasingly solicited in industry.

Although their ability to work for several hours without stopping, the appearance of faults in WMRs is unfortunately inevitable. This can be due to the WMRs components wear, or to their operation conditions which are hard to predict.

According to IFAC SAFEPROCESS technical committee [7], a fault is defined as a non-permitted deviation of at least one system parameter from the acceptable condition, and it can lead to the entire system failure if it is not detected and accommodated in time.

The fault detection and isolation (FDI) allow the system failure avoidance so as to ensure its ability to perform a desired function even in the presence of a fault, either by human intervention or by re-configuring the system control to be tolerant to faults.

In the literature ([12, 19] and references therein), FDI methods are classified into two main categories, namely data-based and model-based methods. These methods can be quantitative (e.g. parity space, observers, ...) or qualitative (e.g. fault trees, digraphs, ...).

One of the most promising techniques for maintaining the system in a normal state when an unexpected fault arises, is fault tolerant control (FTC).

Zhang and Jiang, in [19], define fault tolerant control systems (FTCSs) as systems with the ability to accommodate automatically internal faults. They are able to maintain acceptable performances when a fault occurs. An exhaustive bibliographical review of re-configurable fault tolerant control systems can be found in this same reference.

During mobile robot operation, namely those evolving in a human environment, the slightest fault can lead to potentially high damage that can harm the humans health or the robot’s operating environment. Hence, robots require the ability to tolerate internal faults to continue to perform their tasks autonomously without human immediate intervention.

Regarding WMRs FDI & FTC, several works has been proposed in the literature for either sensor or actuator faults. A state of the art about WMRs FDI & FTC is given in [3]. In [1], some techniques are proposed for WMRs FDI. Authors in [5] propose to use Kalman filter identification technique to detect and isolate WMRs sensor faults, and in [6], a structural analysis-based approach is proposed to detect WMRs actuator faults.

An incorporated adaptive control with navigation functions is proposed to deal with actuator faults of a four-Mecanum WMR in [18].

In [9], several control laws are precomputed to deal with actuator faults of a three omni-directional WMR by considering the robot as redundantly actuated.

In [16], a robust adaptive FTC is proposed for a 4-WMR under actuator faults by varying its center of mass.

In [4], FTC of flat tires is proposed for a 4-wheel skid steering mobile robot. The fault is detected using structural analysis technique, then it is accommodated by correcting the flat tire radius value in the robot angular velocity control.

When a fault occurs either in a sensor or in an actuator, it is generally detected by analyzing the system outputs (measurements) which are returned by sensors. Then, distinguishing sensor from actuator faults is often difficult. Although all...
proposed works in the literature, there is no works treating both WMRs sensor and actuator fault isolation problem.

Our motivation is to treat the WMRs wheel sensor and actuator FDI and fault compensation problem. For that, a model-based method is proposed. The idea is to generate residuals with different signatures under different faults. Then, sensor faults are compensated so as to have the real measurements, and actuator faults are compensated so as to have actuator FTC.

To do, four steps are followed: 1) & 2) sensor and actuator FDI by generating and analyzing residuals using model-based method & extended Kalman filter (EKF), 3) fault estimation by filtering residuals using a Kalman filter (KF), and 4) actuator FTC and sensor faults compensation.

The paper is organized as follows: we firstly point out the unicycle mobile robot kinematic model in section II. Then, the FDI proposed method is explained in Section III. In section IV, a detailed explanation of fault compensation method is given. In section V, simulation results are given for illustration. Finally, some perspectives are given to conclude the article in section VI.

II. UNICYCLE MOBILE ROBOT KINEMATIC MODEL

The robot geometry is presented in Fig. 2. To consider the unicycle mobile robot kinematic model, it is assumed that the robot is placed on a plane surface where $\{0, x, y\}$ is the inertial reference frame and $(G, \overrightarrow{v_x}, \overrightarrow{v_y})$ is a local coordinate frame fixed on the robot at its center of mass $G$.

![Unicycle mobile robot geometry](image)

Fig. 2: Unicycle mobile robot geometry.

The kinematic model is given in the inertial reference by a continuous state-space representation as follows:

$$
\begin{align*}
\dot{x}(t) &= v(t) \cos \theta(t) + q_1(t) \\
\dot{y}(t) &= v(t) \sin \theta(t) + q_2(t) \\
\dot{\theta}(t) &= \omega(t) + q_3(t)
\end{align*}
$$

where, $x$ and $y$ are the robot position along the $x$-axes and the $y$-axes respectively, $\theta$ the robot orientation angle, $v$ the linear velocity associated to the center of mass $G$, $\omega$ the angular velocity of the center of mass, and $q_i(t)$, ($i = 1, 2, 3$) is the model error or uncertainty. Vector $u = [v, \omega]^T$ denotes the input vector (control inputs).

The following assumptions are made:

$(H_1)$ The WMR is equipped with two optical encoders $E_r$ and $E_l$, which return in real time the right and the left wheel angular velocities $\omega_r$ and $\omega_l$ respectively, and with a gyroscope which returns the yaw angle $\theta$ between the mobile robot axle and the $x$-axis.

$(H_2)$ WMR positions along the $x$-axes and the $y$-axes, $x$ and $y$ respectively, are given by a positioning system.

$(H_3)$ In this work, only additive faults are studied.

According to $(H_1)$ and $(H_2)$ assumptions, the measured vector $Y$ is given by:

$$
Y = [x, y, \theta, \omega_r, \omega_l]^T
$$

(2)

The linear and angular velocities $v$ and $\omega$ are given with respect to the angular velocities of the right and the left wheels, (respectively $\omega_r$ and $\omega_l$), according to the following relations: (see [15])

$$
\begin{align*}
\omega &= \frac{(\omega_r - \omega_l)r}{b} \\
v &= \frac{(\omega_r + \omega_l)r}{2}
\end{align*}
$$

(3)

where $r$ and $b$ are two constants denoting respectively the wheel radius and the distance between the two wheels (see Fig. 2). From these two relations, wheel angular velocities $\omega_r$ and $\omega_l$ are given as a function of the control input vector:

$$
\begin{align*}
\omega_r &= \frac{1}{r}v + \frac{b}{2r}\omega_l \\
\omega_l &= \frac{1}{r}v - \frac{b}{2r}\omega_r
\end{align*}
$$

(4)

Finally, the robot model is given by the following state-space representation (SSR):

$$
\begin{align*}
\dot{X} &= F(X, u, t) + q(t) \\
Y &= CX +Du + z(t)
\end{align*}
$$

(5)

where

$$
X = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}, C = \begin{bmatrix} I_{3x3} \\ 0_{2x3} \end{bmatrix}, D = \begin{bmatrix} 1 & b \\ 1 & -\frac{b}{r} \end{bmatrix}
$$

$I_{3x3}$ is a $3 \times 3$ identity matrix and $z(t)$ denotes the measurement noise. Model and measurement noises $q$ and $z$ are supposed to be white noises with known covariance matrices $Q$ and $Z$ respectively. Also, $Q$ is assumed having no cross-correlation with $Z$.

Since measurements are provided at regular times defined by a sampling time $T_s$, it is preferable to discretize the SSR. Such a discretization can be approximated using the Euler’s method, and given by:

$$
\begin{align*}
\dot{X}(k+1) &= X(k) + T_e (F(X, u, k) + q(k)) \\
Y(k) &= CX(k) + Du(k) + z(k)
\end{align*}
$$

(6)

Let’s define the following subsystem containing a subset of the system output:

$$
\begin{align*}
X(k+1) &= X(k) + T_e (F(X, u, k) + q(k)) \\
Y_1(k) &= C_1 X(k) + z_1(k)
\end{align*}
$$

(7)

$C_1 = I_{3x3}$, $Y_1 = X$, and $z_1$ is the $Y_1$ measurements noise. This subsystem (7) will be useful later.
III. Fault Detection and Isolation FDI

Generally speaking, model-based fault detection is linked to residual generation and comparison to predefined thresholds, while the faulty component isolation is based on residual signatures which must be different under different considered faults. A residual is defined according to [7] as a fault indicator, based on a deviation between measurements and model-based estimations.

A. Considered faults

In this work, only wheel sensor and actuator additive (H₃) faults are considered. Gyroscope and positioning system faults are not studied. As faults, are considered: right and left wheel actuator faults, respectively the right encoder, the left encoder, and the gyro- and left optical encoder faults, \( E_r \) and \( E_l \) respectively.

The system (6) in the presence of these sensor and actuator faults is given by:

\[
\begin{align*}
X(k + 1) &= X(k) + T_e F(X, u, k) + E_a f_a + q(k) \\
Y(k) &= CX(k) + E_s f_s + z(k)
\end{align*}
\]

where \( f_a, f_s \) are actuators and sensors additive fault vectors, while \( E_a, E_s \) are respectively actuator and sensor fault matrices. For the considered sensor faults, \( E_s \) is given by:

\[
E_s = \begin{pmatrix} 0_{3x1} & I_{2x1} \end{pmatrix}
\]

While, in this case, the subsystem (7) is given by:

\[
\begin{align*}
X(k + 1) &= X(k) + T_e F(X, u, k) + E_a f_a + q(k) \\
Y_1(k) &= C_1 X(k) + z_1(k)
\end{align*}
\]

B. Residual generation and signature

The goal is to generate residuals with different signatures under the different considered faults, to insure the isolation.

Three residuals are generated as follows:

\[
\begin{align*}
  r_1 &= \omega_{r,\text{measured}} - \dot{\omega}_r \\
  r_2 &= \omega_{l,\text{measured}} - \dot{\omega}_l \\
  r_3 &= \theta_{\text{measured}} - \dot{\theta}
\end{align*}
\]

where, \( \omega_{r,\text{measured}}, \omega_{l,\text{measured}}, \theta_{\text{measured}} \) denote respectively, the right encoder, the left encoder, and the gyroscope measurements (see Fig. 3).

[C 1]

Given by 10, one obtains:

\[
\begin{align*}
  \dot{X}(k+1|k) = \dot{X}(k|k) + T_e F(\dot{X}, u, k) \\
  P(k+1|k) = A_1(k) P(k|k) A_1^T(k) + Q
\end{align*}
\]

where \( P(k+1|k) \) denotes the a priori error covariance matrix associated to the state vector \( X \) and \( A_1 \) the linearized matrix of function \( F \) with \( A_1(k) = I + T_e \frac{\partial F}{\partial X} \) where

\[
\frac{\partial F}{\partial X} = \begin{bmatrix} \frac{\partial F_1}{\partial x_1} & \frac{\partial F_1}{\partial x_2} & \ldots & \frac{\partial F_1}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial F_n}{\partial x_1} & \frac{\partial F_n}{\partial x_2} & \ldots & \frac{\partial F_n}{\partial x_n} \end{bmatrix} \| \dot{X}(k|k) \|
\]

and \( I \) denotes the identity matrix with the appropriate dimension. The correction step contains three equations:

\[
\begin{align*}
  \hat{X}(k+1|k+1) &= \hat{X}(k+1|k) + K(k+1) \times \\
  &\dot{Y}_1(k+1) - C_1 \dot{X}(k+1|k) \\
  K(k+1) &= P(k+1|k+1) C_1^T \left( C_1 P(k+1|k+1) C_1^T + Z_1 \right)^{-1} \\
  P(k+1|k+1) &= (I - K(k+1) C_1) P(k+1|k)
\end{align*}
\]

\( K(k+1) \) the filter gain matrix. Note that, the estimation error for the system (7) in the presence of the considered sensor and actuator faults is given by:

\[
e(k+1) = X(k+1) - \hat{X}(k+1) = (X(k) + T_e F(X, u, k) + E_a f_a + q(k)) - (\hat{X}(k) + T_e F(\hat{X}, u, k) + K(k+1) [\dot{Y}_1(k+1) - C_1 \dot{X}(k+1|k)])
\]

with \( Y_1 \) given by 10, one obtains:

\[
e(k+1) = (1 + KC_1) e(k) + T_e F(X, u, k) - F(\hat{X}, u, k) + K z_1 + q + E_a f_a
\]

As it can be noticed in (16), sensor faults do not appear in the estimation error for the system (7), namely \( \hat{\theta} \) is influenced just by actuator faults. This allows to isolate sensor and actuator faults (see the section hereafter).
D. Residual signatures

In the absence of faults, residuals in (11) are close to zero, while in the occurrence of a component fault, residuals being sensitive to such a fault get away from zero. To detect faults, fixed thresholds $\pm r_{th,i}$ are defined in the nominal operation mode using three-sigma method [13] for each residual. The principle is to detect a fault if a residual exceeds its predefined threshold. Using the three-sigma method, false alarms (i.e. detecting a fault in a healthy mode) can be detected, and hence, they must be tolerated. To do, a fault detection is handled when the threshold is exceeded more than $N$ consecutive times.

The decision residual $R_i$ is turned into 1 when it satisfies the fault handling condition as follows:

$$R_i(k) = \begin{cases} 1 & \text{if } (r_i(j) < -r_{th,i} \text{ or } r_{th,i} < r_i(j)), \\ \forall j \in [k - N + 1, k] & \text{0 otherwise} \end{cases}$$

$\pm r_{th,i}$ are fixed thresholds. Table I illustrates the three residual signatures under the considered faults.

<table>
<thead>
<tr>
<th>Fault Res.</th>
<th>$E_r$</th>
<th>$E_i$</th>
<th>$W_r$</th>
<th>$W_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R_1$</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$R_2$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$R_3$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

TABLE I: Residual signatures under wheel sensor & actuator faults.

IV. FAULT ESTIMATION & ACCOMMODATION

A. Fault estimation (FE)

The generated residuals are the difference between the robot measurements (in simulation the kinematic model outputs, for which noises are added randomly to get closer to reality), and output estimations, and calculations using mathematical relations (see (11)).

To enhance the FDI time, measurements noises are filtered from residuals (11). Filtering residuals allows to detect and isolate faults more quickly (see Fig. 6 in section V).

Added to that, residuals $r_1$ and $r_2$ are respectively the image of right and left wheel sensor and actuator faults. Filtering noise from these residuals allows a perfect fault estimation (see Fig. 8, section V).

In this work a Kalman filter (KF) is used to filter noises from residuals (see Fig. 4). The following linear system is defined:

$$\begin{align*}
X_r(k + 1) &= A_r X_r(k) + q_r, \\
Y_r(k) &= C_r X_r(k) + z_r,
\end{align*}$$

with $X_r = [r_1, r_2, r_3]^T$ the state vector with residuals, $Y_r$ the output vector, $A_r$ and $C_r$ are $(3 \times 3)$ identity matrices. $q_r$ and $z_r$ denote the model and measurement noises. They have the same statistical properties as $q$ and $z$, and have as covariance matrices respectively $Q_r$ and $Z_r$. $R_r$ matrix values correspond on the residual variances in the nominal mode, while $Q_r$ values must respect the compromise between filtering and delay.

B. Fault compensation & accommodation

The objective here is to detect and distinguish wheel sensor faults from wheel actuator faults, then to compensate these latter so as the robot accomplishes its mission despite the faulty mode.

For simplification reasons, the robot is controlled to reach a desired position $[x_d, y_d]^T$ starting from its initial position $[x_0, y_0]^T$. It follows a control law depending on the distance between the current and the desired positions (see (2)):

$$\begin{align*}
v &= k_v \sqrt{(x_d - x)^2 + (y_d - y)^2} \\
\omega &= k_w \arctan \left( \frac{y_d - y}{x_d - x} \right)
\end{align*}$$

(18)

where $k_v$ and $k_w$ are two tuning control parameters.

The fault accommodation principle is defined as follows: When a fault occurs in $E_r$ or $E_i$, it is compensated in the angular velocities measurement, $\omega_r$ or $\omega_l$ (returned by the faulty right or left encoder) to get the real and correct wheels angular velocities $\omega^*_r$ or $\omega^*_l$, as follows: (see Fig. 5).

$$\begin{align*}
\omega^*_r &= \omega_r - r_{1, \text{filtered}} \\
\omega^*_l &= \omega_l - r_{2, \text{filtered}}
\end{align*}$$

(19)

On the other hand, when a fault occurs in the right or the left wheel actuators, the robot position $x$ and $y$ is influenced (i.e. the robot gets away from the desired position). The control inputs resulting from the robot faulty position (see (18)) is hence incorrect. To accommodate actuator faults, new control input $[v^*, w^*]^T$ is calculated by compensating the estimated faults as follows: (see Fig. 5).

$$\begin{align*}
v^* &= v - \frac{2}{b} (r_{1, \text{filtered}} + r_{2, \text{filtered}}) \\
w^* &= \omega - \frac{2}{b} (r_{1, \text{filtered}} - r_{2, \text{filtered}})
\end{align*}$$

(20)

The following algorithm resumes both wheel sensor and actuator compensation principle, according to table I:

- if $R_i = 1, (i = 1 \text{ or } i = 2)$ and $R_3 = 0$, fault is occurred on $E_r$ ($i = 1$) or on $E_i$ ($i = 2$). Faulty encoder measurements are then corrected using (19), so that real wheel velocities are displayed.
- if $R_i = 1, (i = 1 \text{ or } i = 2)$ and $R_3 = 1$, fault is occurred on $W_r$ ($i = 1$) or $W_l$ ($i = 2$). Actuator fault is then compensated using (20), so as to have a new control inputs.

V. SIMULATION RESULTS

Let’s recall that in this work, only additive faults are considered (\(\mathcal{H}_3\)). They are simulated as a constant bias (step) or as a time varying ramp signals, which can represent component wear. The challenge lies in completely isolating and accommodating wheel sensor and actuator additive faults.
whatever their type or magnitude. For simulations, it is assumed that when the mobile robot starts to function, all its components are in a normal mode, and when a fault occurs, the system remains in the faulty state (permanent faults).

For the wheel radius $r$ and the distance between the wheels $b$, following values are taken: $r = 0.095m$, $b = 0.33m$. They correspond on the Pionner-3DX parameters.

Only sensitive residuals for each simulated fault are presented, the other residuals are close to zero, hence, they are not presented.

In Fig. 6, a fault with a magnitude of $10^{-2}$ rad/s is simulated in $E_r$. The figure shows the fault detection using residual $r_1$, respectively before and after filtering. Results show that with the non-filtered $r_1$, the fault took 0.53s to be detected while with the filtered one, it took just 0.19s. In the following results, only filtered residuals are presented.

Figure 7 illustrates a FDI in actuator $W_l$, by the two residuals $r_2$ and $r_3$. The simulated fault magnitude is $-10^{-2}$ rad/s. This represents a wear of the right actuator in the order of 3.6 rad/hour. The fault is detected 0.04s after its apparition by $r_3$, and isolated 0.06s after by $r_2$.

In Fig. 8, $W_l$ fault is plotted without encoder fault (in green), then with a non compensated (red signal) and compensated $E_l$ fault (blue) in the same figure to illustrate sensors faults accommodation principle.

Figure 10 illustrates the robot paths in nominal functioning mode with respect to the desired ones according to x and y axes. Without any fault, the robot follows the desired path. In Fig. 11, each of the WMR controlled with the proposed FTC, and the WMR controlled with the non FTC paths are presented with respect to the desired path according to x and y axes in presence of a $W_r$ actuator fault. The latter is a time varying ramp, with a magnitude of $-10^{-3}$ rad/s and time apparition of 7s. The figure shows clearly that the proposed method accommodates perfectly the fault. The fault tolerant robot path is the same as that of the non-faulty one presented in Fig. 10. Finally, in Fig. 12, a fault in the $W_l$ actuator is simulated at time 7s. The latter is a step with a magnitude of $-1 rad/s$. The result of compensating the fault perfectly by the proposed approach is evident since the fault estimation method is robust.
VI. Conclusion

In this article, WMRs wheel actuator and sensor faults are studied. The goal is first to detect and isolate, then to accommodate faults as early as possible for insuring the functionality of the unicycle WMR despite the presence of faults. The proposed approach relies on generating and analyzing residuals. Then, residuals are filtered using a KF so as to accelerate the FDI and estimate the faults. Faults are then compensated to ensure the robot accomplishes its mission despite the degraded mode. The strong point of the proposed approach is that it can accommodate additive faults whatever their type as long as it lies on its estimation and compensation. In the future, the proposed method will be applied on an OMRON unicycle WMR to be validated. The method can also be applied on a 4 skid steering mobile robot. The latter can be modeled as a unicycle WMR [4].

References


