A Dynamic Grid-based Q-learning for Noise Covariance Adaptation in EKF and its Application in Navigation

Xiang Dai, Hassen Fourati, Christophe Prieur

A Dynamic Grid-based Q-learning for Noise Covariance Adaptation in EKF and its Application in Navigation

Xiang Dai[1], Hassen Fourati[1] and Christophe Prieur[2]

***Abstract*— The process and measurement noise covariance matrices significantly impact the Extended Kalman Filter (EKF) performance and are often hand-tuned in practice, which usually entails a tedious task. Q-learning, a well-known method in reinforcement learning, has been applied recently to better adapt the noise covariance matrices for the EKF, thanks to its simplicity and capability in handling uncertain environments. Typically, some heuristics are involved in designing the Q-learning-based EKF (QLEKF), such as tuning grid size and covariance matrices values of each state, which inevitably degrades the estimation performance when the heuristics are not suitable. We propose a dynamic grid-based Q-learning EKF (DG-QLEKF) to overcome that drawback, which brings two novelties, an updated ϵ-greedy algorithm and a dynamic grid strategy. The proposed algorithm and strategy can thoroughly exploit arbitrary search scope and find appropriate values of noise covariance matrices. The effectiveness of DG-QLEKF, applied in navigation for attitude and bias estimation, is validated through the Monte Carlo method and real flight data from an unmanned aerial vehicle. The DG-QLEKF leads to much more improved state estimation than the QLEKF and traditional EKF.**

I. INTRODUCTION

The Kalman filter (KF) is known to be optimal in minimizing the estimated error covariance [1], and in separating stochastic noises from real signals given sensors measurements for linear systems [2]. For systems with nonlinear dynamic models, the EKF linearizes dynamics and output functions at the current state estimate to propagate predictor-corrector functions of KF. To date, the EKF has been extensively used in state estimation under noisy data and applied to engineering domains, e.g., navigation [3], robotics, computer vision [4], and electrical power systems [5]. It is well known that the process and measurement noise covariance matrices significantly impact the EKF performance. In the absence of exact statistical knowledge on noises, assigning them with appropriate values often requires a tedious task. In practice, the noise covariance matrices are commonly hand-tuned in the EKF through trials and errors or defined by empirical rules, which can sometimes be time-consuming or results in barely satisfactory performance.

As one of the most important reinforcement learning methods, Q-learning [6] has drawn increasing interest in adapting the noise covariance matrices of EKF [7], [8], for its model-free algorithm, low computation demand, and capability in achieving optimality in Markov decision processes [9]. In our previous work [10], a Q-learning-based EKF (QLEKF) is proposed to autonomously adapt the values of process and measurement noise covariance matrices in the attitude estimation of a rigid body. Though improvement is revealed in estimation errors compared to the traditional EKF using hand-tuned noise covariance matrices, the design of QLEKF involves certain amounts of heuristics and a rule of thumb.

This paper proposes a dynamic grid-based Q-learning EKF (DG-QLEKF) to address the deficiencies mentioned above in QLEKF. To begin with, we propose an updated ϵ-greedy algorithm that enables unbiased exploration of state-action space and non-worse state convergence w.r.t. the reference state. Next, we propose the dynamic 3-by-3 grid determined by the dynamic center and ratio, where the center tracks the best-found noise covariance matrices, and the ratio defines the searching scope. In the continuation, we introduce the absolute accumulative innovation term as the criterion to quantify the performance of each process and measurement covariance matrices pair.

The main contributions of the paper are threefold. First, we propose an updated ϵ-greedy algorithm suited for optimal value searching in the Q-learning context. Second, we propose an efficient dynamic grid strategy for Q-learning with adaptable update characteristics that save the effort in defining the grid. Third, we propose an easy-to-implement and quick-responsive Q-learning algorithm that can adapt the noise covariance matrices of the EKF for large scope and with fine precision.

The rest of the paper is organized as follows. Section II formulates the state-space dynamic functions for nonlinear systems and introduces the traditional EKF. Section III recalls some preliminaries on the Q-learning and QLEKF. Section IV details the dynamic grid strategy and the updated ϵ-greedy algorithm, followed by the overall DG-QLEKF. Monte Carlo simulations are performed in Section V to estimate the attitude of a rigid body and sensor biases using the DG-QLEKF. Conclusion and future works are given in Section VI.

II. STATE-SPACE REPRESENTATION AND EKF-BASED ESTIMATION ALGORITHM

In this section, we formulate the dynamic functions of the nonlinear system and introduce the traditional EKF to estimate the observable states.

[1]Xiang Dai and Hassen Fourati are with GIPSA-Lab, Inria, CNRS, Univ. Grenoble Alpes, Grenoble INP, Grenoble, 38400, France, xiang.dai@gipsa-lab.grenoble-inp.fr, hassen.fourati@gipsa-lab.grenoble-inp.fr

[2]Christophe Prieur is with GIPSA-Lab, CNRS, Univ. Grenoble Alpes, Grenoble INP, Grenoble, 38400, France, christophe.prieur@gipsa-lab.fr

This work has been partially supported by MIAI@Grenoble Alpes (ANR-19-P3IA-0003).

A. System formulation

We consider a discrete Markov model to describe a nonlinear system, with two dynamic functions as follows

$$\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k) + \boldsymbol{w}_k, \quad (1)$$

$$\boldsymbol{y}_k = g(\boldsymbol{x}_k) + \boldsymbol{v}_k, \quad (2)$$

where $f(\cdot)$ and $g(\cdot)$ are process and measurement dynamic functions, respectively, and are assumed to be continuously differentiable, $\boldsymbol{x}_k$ is the system state to be estimated, $\boldsymbol{y}_k$ is the system measurable output, $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ are the process and measurement noises, respectively.

We assume that the process and measurement noises are independent Gaussian with zero mean, which means for $\forall\ i, j = 0, 1, 2, ...,$ we have $\mathbb{E}[\boldsymbol{w}_i] = \boldsymbol{0}$, $\mathbb{E}[\boldsymbol{v}_i] = \boldsymbol{0}$, $\mathbb{E}[\boldsymbol{w}_i \boldsymbol{w}_i^T] = \boldsymbol{Q}$, $\mathbb{E}[\boldsymbol{v}_i \boldsymbol{v}_i^T] = \boldsymbol{R}$, $\mathbb{E}[\boldsymbol{w}_i \boldsymbol{v}_j^T] = \boldsymbol{0}$, for $i \neq j$: $\mathbb{E}[\boldsymbol{w}_i \boldsymbol{w}_j^T] = \boldsymbol{0}$ and $\mathbb{E}[\boldsymbol{v}_i \boldsymbol{v}_j^T] = \boldsymbol{0}$, where $\boldsymbol{Q}$ and $\boldsymbol{R}$ are the process and measurement noises covariance matrix, respectively.

B. The traditional extended Kalman filter

For nonlinear systems expressed by (1) and (2), the EKF used for state estimation is summarized in Alg. 1.

Algorithm 1 Traditional EKF

Input $\hat{\boldsymbol{x}}_{k|k}, \boldsymbol{P}_{k|k}, \boldsymbol{y}_k, \boldsymbol{Q}, \boldsymbol{R}$
$\hat{\boldsymbol{x}}_{k+1|k} = f(\hat{\boldsymbol{x}}_{k|k}, \boldsymbol{0})$
$\boldsymbol{A}_k = \frac{\partial f}{\partial \boldsymbol{x}}(\hat{\boldsymbol{x}}_{k|k})$
$\boldsymbol{P}_{k+1|k} = \boldsymbol{A}_k \boldsymbol{P}_{k|k} \boldsymbol{A}_k^T + \boldsymbol{Q}$
$\boldsymbol{C}_{k+1} = \frac{\partial g}{\partial \boldsymbol{x}}(\hat{\boldsymbol{x}}_{k+1|k})$
$\boldsymbol{K}_{k+1} = \boldsymbol{P}_{k+1|k} \boldsymbol{C}_{k+1}^T (\boldsymbol{C}_{k+1} \boldsymbol{P}_{k+1|k} \boldsymbol{C}_{k+1}^T + \boldsymbol{R})^{-1}$
$\tilde{\boldsymbol{y}}_{k+1} = \boldsymbol{y}_{k+1} - g(\hat{\boldsymbol{x}}_{k+1|k}, \boldsymbol{0})$
$\hat{\boldsymbol{x}}_{k+1|k+1} = \hat{\boldsymbol{x}}_{k+1|k} + \boldsymbol{K}_{k+1} \tilde{\boldsymbol{y}}_{k+1}$
$\boldsymbol{P}_{k+1|k+1} = \boldsymbol{P}_{k+1|k} - \boldsymbol{K}_{k+1} \boldsymbol{C}_{k+1} \boldsymbol{P}_{k+1|k}$
Output $\hat{\boldsymbol{x}}_{k+1|k+1}, \boldsymbol{P}_{k+1|k+1}, \tilde{\boldsymbol{y}}_{k+1}$

III. THE Q-LEARNING-BASED APPROACH FOR NOISE COVARIANCE ADAPTATION

In this section, the Q-learning basics and their combination with the EKF are introduced, and then our previous work [10] on Q-learning-based adaptation is recalled.

A. Preliminaries on Q-learning

Q-learning is a reinforcement learning method that maximizes the long-term reward in a multi-state environment. That environment typically consists of discrete state-action pairs, each assigned with a scalar called Q-value. In Q-learning, the agent attempts to obtain an optimal sequential actions decision by maximizing the Q-value of each state-action pair, called exploitation. Visiting the non-exploited state-action space by the agent is called exploration. Usually, exploitation and exploration need to be balanced by an action strategy. The ϵ-greedy algorithm [11] is adopted in this work for its implementation simplicity and effectiveness compared to other exploration methods, e.g., random walk exploration [12], [13], and Softmax action selection method [14]. In the ϵ-greedy algorithm, the agent chooses a random action with a predefined probability of ϵ or picks the action that maximizes the Q-value with the probability of $1 - \epsilon$.

Each time after executing an action a, the agent receives a response from the environment, which is translated to a reward (R), showing how good the action is. The cumulative reward is stored as Q-value

$$Q(s, a) = (1 - \alpha)Q(s, a) + \alpha[R + \gamma \max_{a \in A(s)} Q(s', a)], \quad (3)$$

where $Q(s, a) \in \mathbb{R}$ is the Q-value for the action a in state s, $R \in \mathbb{R}$ is the reward gained by executing a in state s, α is the learning rate, γ is the discount factor, and $A(s)$ is the possible actions set when the agent is at s.

B. Q-learning-based noise covariance adaptation approach

In the Q-learning method, an agent typically moves among discrete states in a grid. For example, we can set the noise covariance matrices to M different values for $\boldsymbol{Q}$ as $\{\boldsymbol{Q}^{(1)}, ..., \boldsymbol{Q}^{(M)}\}$, and N different values for $\boldsymbol{R}$ as $\{\boldsymbol{R}^{(1)}, ..., \boldsymbol{R}^{(N)}\}$, and place them in a M-by-N grid, of which the element (i, j) stands for the noise covariance matrices $(\boldsymbol{Q}^{(i)}, \boldsymbol{R}^{(j)})$. In this way, $(\boldsymbol{Q}^{(i)}, \boldsymbol{R}^{(j)})$ are used in the EKF when the agent is at state (i, j).

In [10], we have proposed the QLEKF that runs three parallel EKFs at each time step: the traditional EKF, which sets some initial values of $(\boldsymbol{Q}, \boldsymbol{R})$ for all time steps and serves as the benchmark for Q-learning; the learning EKF, which searches appropriate noise covariance matrices from the grid by the Q-learning algorithm; and the learned EKF, which outputs the result of estimation according to the covariance matrices found by the learning EKF. Please refer to Alg. 2 in [10] for more details about the QLEKF. As shown in [10], the QLEKF exhibits the benefit of improving the EKF state estimation by searching for more appropriate noise covariance matrices from a predefined set of values. However, QLEKF contains potential insufficiencies:

1) Heuristics are used to define $(\boldsymbol{Q}, \boldsymbol{R})$ for the traditional (reference) EKF, which matters to a great extent to the QLEKF estimation performance as it determines the reward computation.
2) The determination of the grid size and the value of each $(\boldsymbol{Q}^{(i)}, \boldsymbol{R}^{(j)})$ is generally heuristic, which is indeed a key factor for the QLEKF performance.
3) The state estimation produced by the learned EKF uses the exact sequence $\{(\boldsymbol{Q}^{(i)}, \boldsymbol{R}^{(j)})\}$ visited by the learning EKF, in which the random actions may degrade the estimation performance.

IV. THE DYNAMIC GRID-BASED Q-LEARNING ALGORITHM APPLIED TO THE EKF

To overcome the potential drawbacks of the QLEKF listed in Subsection III-B, we propose a deterministic way to design the dynamic grid and an update of the ϵ-greedy algorithm.

A. The dynamic grid and updated ϵ-greedy algorithm

We propose an advanced variant of the ϵ-greedy algorithm and the way to build the dynamic grid, shown in Alg. 2. First, Steps 4-5 enable an unbiased exploration at state s when its possible actions possess the same Q-value. Otherwise, if exploitation (Step 7) is enforced in that case, the agent would choose the first action located in $A(s)$ by default, which causes a biased exploration. Second, Steps 8-9 ensure that only the states with positive reward can be chosen to stay, which guarantees the dominance of noise covariance matrices visited in Q-learning over their reference values.

Algorithm 2 Updated ϵ-greedy Algorithm

```
1:  a = Updated ε-greedy(ε, s, A(s), Q(s, a))
2:  Input: s, A(s) and Q(s, a)
3:  Generate n from uniform distribution n ~ U(0, 1)
4:  if n < ε or ∀a_1, a_2 ∈ A(s), Q(s, a_1) = Q(s, a_2) then
5:      select a randomly from A(s)
6:  else
7:      a = arg max_{a∈A(s)} Q(s, a)
8:      if a='stay' and Q(s, 'stay') < 0  then
9:          select action a randomly from A(s) \ 'stay'
10:     end if
11: end if
```

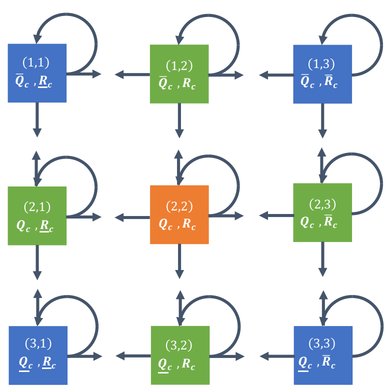


Fig. 1: The 3-by-3 dynamic grid of states (process and measurement noises covariance matrices pairs) and the corresponding possible actions, represented by arrows. The agent is allowed to '*stay*' at its current state or move to its adjacent state by executing one action.

In what follows, we propose a dynamic grid strategy to address the difficulty in determining the grid size and state values of the QLEKF. Let $(\boldsymbol{Q}_0, \boldsymbol{R}_0)$ denote the initial noise covariance matrices used for the traditional EKF. We consider that the adaptation of noise covariance matrices $(\boldsymbol{Q}, \boldsymbol{R})$ is carried out in a set as $C = \{(\boldsymbol{Q}, \boldsymbol{R}) \mid \boldsymbol{Q} = q_0\boldsymbol{Q}_0, \boldsymbol{R} = r_0\boldsymbol{R}_0, 0 < \underline{q} \leq q_0 \leq \overline{q}, 0 < \underline{r} \leq r_0 \leq \overline{r}\}$, where q_0 and r_0 are the multiplication ratio associated with $\boldsymbol{Q}_0$ and $\boldsymbol{R}_0$, respectively. The dynamic grid shown in Fig. 1 is created in the following way: i), determine $(\boldsymbol{Q}_c, \boldsymbol{R}_c) \in C$ for the center element; ii), compute $\underline{\boldsymbol{Q}} = \max\{\underline{q}, \frac{1}{q}\}\boldsymbol{Q}_c$, $\overline{\boldsymbol{Q}} = \min\{\overline{q}, q\}\boldsymbol{Q}_c$, $\underline{\boldsymbol{R}} = \max\{\underline{r}, \frac{1}{r}\}\boldsymbol{R}_c$, $\overline{\boldsymbol{R}} = \min\{\overline{r}, r\}\boldsymbol{R}_c$ where $q > 1$ and $r > 1$ are the multiplication factors associated with $\boldsymbol{Q}_c$ and $\boldsymbol{R}_c$, respectively.

We design the 3-by-3 dynamic grid for the following reasons. First, 3-by-3 is the smallest grid that contains the noise covariance matrices combinations of reference values $\boldsymbol{Q}_c$, $\boldsymbol{R}_c$ and their greater and smaller counterparts: $\underline{\boldsymbol{R}}$, $\overline{\boldsymbol{R}}$, $\underline{\boldsymbol{Q}}$ and $\overline{\boldsymbol{Q}}$. Second, it is sufficient for Q-learning to explore and exploit all directions from the center element. Third, its all 9 elements can be visited whit a shorter learning period, compared to larger size grids in [8], [10]. Fourth, by dynamically updating the ratio r, q and the central element $(\boldsymbol{Q}_c, \boldsymbol{R}_c)$, it allows Q-learning to search in an arbitrarily large scope (by manipulating $\underline{q}$, $\underline{r}$, $\overline{r}$ and $\overline{q}$) with an arbitrarily high precision (by setting r and q close to 1).

B. The Dynamic Grid-based Q-learning EKF approach

The DG-QLEKF is summarized in Alg. 3, in which Step 18 integrates Alg. 4 to update the 3-by-3 dynamic grid. We emphasize the difference in contrast to the QLEKF in elaborating the DG-QLEKF. To begin with, we introduce an absolute metric T in Step 11 as the innovation term norm of the learning EKF, which eliminates the impact of the innovation term of the reference EKF and reflects the absolute innovation magnitude of each distinct $(\boldsymbol{Q}, \boldsymbol{R})$. If Step 17 in Alg. 3 is satisfied, then we focus on T_{mean}, the average T of the state s_c as presented in Step 2 of Alg. 4, because it further mitigates the impact of previous state estimate $\hat{\boldsymbol{x}}_k^l$ on T and indicates the average innovation term magnitude when *'stay'* action is executed. If a lower T_{mean} appears, we first reset n_T, the counter of local state convergence without finding a lower T_{mean}, to 0, making the current grid be continually examined enough times before update. Then if s_c corresponds to the non-center element, we move the center to that element and reset the ratio to initial values as in Step 6, which generates a new 3-by-3 grid. If no lower T_{mean} is observed, n_T is self added by 1 until it reaches the predefined threshold n_{ratio}, in which case we increase r and q. The logic behind this is, first, multiple times (n_{ratio}) check after local convergence averts the non-fully visited situation. Second, every newly found center is an appropriate values candidate. In turn, from its near (small ratio) to distant (large ratio) neighbors, its T_{mean} should be compared with that of other candidates. Unlike in the QLEKF, where the traditional (reference) EKF uses fixed noise covariance matrices, in the DG-QLEKF, each time $(\boldsymbol{Q}_c, \boldsymbol{R}_c)$ is changed (lower T_{min} is found), it will be applied to the traditional EKF from the next period learning, ensuring that the best-found noise covariance matrices are referred to compute the reward R in Step 10 of Alg. 3. After the Q-learning has searched throughout the predefined scope (Step 23 of Alg. 3), the noise covariance matrices $(\boldsymbol{Q}_c, \boldsymbol{R}_c)$ that produce the lowest average innovation term, are used as deterministic values of noise covariance matrices to compute the final state estimation.

Algorithm 3 Dynamic Grid Q-learning Extended Kalman Filter (DG-QLEKF)

```
Initialize x̂^t_{0|0} = x̂^l_{0|0}, P^t_{0|0} = P^l_{0|0}, r_, r_0, q_, r̄, q̄, q_0, R_0, Q_0, n_l and n_ratio
R ← 0, T ← 0, T_record ← ∅, T_min ← ∞, k ← 0, R_c ← R_0, Q_c ← Q_0, r ← r_0, q ← q_0, n_T ← 0
∀s ∈ S, a ∈ A(s), Q(s, a) ← 0
repeat
    a = Updated ε-greedy(0.1, s, A(s), Q(s, a))
    Execute action a and obtain state s'
    for each time step in one period do
        [x̂^t_{k+1}, P^t_{k+1}, ỹ^t_{k+1}] = EKF(x̂^t_k, P^t_k, y^t_k, Q_c, R_c) {Reference EKF}
        [x̂^l_{k+1}, P^l_{k+1}, ỹ^l_{k+1}] = EKF(x̂^l_k, P^l_k, y^l_k, Q(s'), R(s')) {Learning EKF}
        R ← R + ||ỹ^t_{k+1}|| − ||ỹ^l_{k+1}||
        T ← T + ||ỹ^l_{k+1}|| {accumulate absolute innovation term corresponds to (Q(s'), R(s'))}
        k ← k + 1
    end for
    Update Q(s, a) by (3)
    T_record ← [T_record, T], s ← s', R ← 0, T ← 0
    x̂^l_{k+1} ← x̂^t_{k+1}, P^l_{k+1} ← P^t_{k+1}
    if a state s_c appears sufficiently frequent in the latest n_l consecutive states[1] then
        Update Q_c, R_c, q and r by DGU (Alg. 4)
        T_record ← ∅ {reset record lists after convergence}
        select s randomly state from S
        ∀s ∈ S, a ∈ A(s), Q(s, a) ← 0
    end if
until q = q̄, r = r̄ and n_T = n_ratio
use Q_c and R_c to compute {x̂_k} and {P_k}
```

Algorithm 4 Dynamic Grid Update (DGU)

```
Input: s_c, T_record, q, r, T_min, n_l and the current 3-by-3 grid
T_mean ← mean value of T corresponding to s_c over the latest n_l elements of T_record
if T_mean < T_min {better innovation found for converged state} then
    n_T ← 0 {reset no-new-better counter}
    if s_c ≠ (2, 2) {new center is found} then
        Q_c, R_c ← noise covariance matrices values in the current 3-by-3 grid corresponding to s_c
        r ← r_0, q ← q_0 {reset the ratio}
        T_min ← T_mean
    end if
else
    n_T ← n_T + 1
    if n_T = n_ratio then
        increase r and q {current grid is well exploited, expand search scope}
    end if
end if
```

C. Estimation error bounds of DG-QLEKF

In this subsection, we prove that the mean square of the state estimation error of DG-QLEKF is exponentially bounded.

According to Alg. 1, (1) and (2) can be expanded at $\hat{\boldsymbol{x}}_k$ $\hat{\boldsymbol{x}}_{k+1|k}$ using Taylor approximation as

$$\begin{aligned}\boldsymbol{x}_{k+1} =& f(\hat{\boldsymbol{x}}_k) + \boldsymbol{A}_k(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k) + \boldsymbol{w}_k + \phi(\boldsymbol{x}_k, \hat{\boldsymbol{x}}_k), \quad (4)\\ \boldsymbol{y}_{k+1} =& g(\hat{\boldsymbol{x}}_{k+1|k}) + \boldsymbol{C}_k(\boldsymbol{x}_{k+1} - \hat{\boldsymbol{x}}_{k+1|k}) + \boldsymbol{v}_k + \\ & \psi(\boldsymbol{x}_{k+1}, \hat{\boldsymbol{x}}_{k+1|k}), \quad (5)\end{aligned}$$

where $\phi(\cdot)$ and $\psi(\cdot)$ are nonlinear functions to offset the linearization error of $f(\cdot)$ and $g(\cdot)$, respectively.

Let the state estimation error is defined by $\tilde{\boldsymbol{x}}_k = \boldsymbol{x}_k - \hat{\boldsymbol{x}}_k$. Combining Step 2, 7, 8 of Alg. 1, (4) and (5), we have

$$\tilde{\boldsymbol{x}}_{k+1} = (\boldsymbol{I} - \boldsymbol{K}_{k+1}\boldsymbol{C}_{k+1})\boldsymbol{A}_k\tilde{\boldsymbol{x}}_k + \boldsymbol{r}_k + \boldsymbol{s}_k, \quad (6)$$

where $\boldsymbol{r}_k = (\boldsymbol{I} - \boldsymbol{K}_{k+1}\boldsymbol{C}_{k+1})\phi(\boldsymbol{x}_k, \hat{\boldsymbol{x}}_k) - \boldsymbol{K}_{k+1}\psi(\boldsymbol{x}_{k+1}, \hat{\boldsymbol{x}}_{k+1|k})$, $\boldsymbol{s}_k = (\boldsymbol{I} - \boldsymbol{K}_{k+1}\boldsymbol{C}_{k+1})\boldsymbol{w}_k - \boldsymbol{K}_{k+1}\boldsymbol{v}_{k+1}$.

Theorem 1: *Consider a nonlinear stochastic system given by* (1), (2). *If there are real numbers* $\underline{a}, \overline{a}, \underline{c}, \overline{c}, \underline{p}, \overline{p}, \underline{q}', \overline{q}', \underline{r}', \overline{r}', \alpha_1, \alpha_2, \alpha_3 > 0$, *such that for every* $k \geq 0$ *we have*

$$\underline{a} \leq ||\boldsymbol{A}_k|| \leq \overline{a}, \underline{c} \leq ||\boldsymbol{C}_k|| \leq \overline{c}, ||\tilde{\boldsymbol{x}}_k|| \leq \alpha_1, \quad (7)$$

$$\underline{p}\boldsymbol{I} \leq \boldsymbol{P}_k \leq \overline{p}\boldsymbol{I}, \underline{q}'\boldsymbol{I} \leq \boldsymbol{Q}_c \leq \overline{q}'\boldsymbol{I}, \underline{r}'\boldsymbol{I} \leq \boldsymbol{R}_c \leq \overline{r}'\boldsymbol{I}, \quad (8)$$

$$||\phi(\boldsymbol{x}_k, \hat{\boldsymbol{x}}_k)|| \leq \alpha_2||\tilde{\boldsymbol{x}}_k||^2, \quad (9)$$

$$||\psi(\boldsymbol{x}_{k+1}, \hat{\boldsymbol{x}}_{k+1|k})|| \leq \alpha_3||\tilde{\boldsymbol{x}}_k||^2, \quad (10)$$

then we can design a $\epsilon_0 > 0$ *provided that* $||\tilde{\boldsymbol{x}}_0|| \leq \epsilon_0$ *such that the estimation error* $\tilde{\boldsymbol{x}}_k$ *of DG-QLEKF represented by Alg. 3 is bounded in mean square and bounded with probability one.*

V. NUMERICAL SIMULATIONS WITH REAL DATA

We consider the framework of navigation systems and focus on the problem of attitude/bias estimation using magnetic, angular rate, and gravity (MARG) sensors. The proposed estimation approach is evaluated with 50 Monte Carlo simulations using real flight data from Euroc database [15].

A. Dynamic model formulation for attitude and bias

We build up the dynamic model for a rigid body moving in space. Let $\mathcal{B}$ denote the body frame and $\mathcal{N}$ denote the navigation frame. First, we introduce the dynamic of attitude as $\dot{\boldsymbol{q}} = \frac{1}{2}\boldsymbol{\Omega}(\boldsymbol{\omega})\boldsymbol{q}$, where $\boldsymbol{q} = [q_0, q_1, q_2, q_3]^T$ is the quaternion that describes the attitude with $\|\boldsymbol{q}\| = 1$, of which q_0 is the scalar part and q_1, q_2, q_3 compose the vector part,

[1] In this case, we consider $\max_{s \in S, a \in A(s)} Q(s, a) = Q(s_c, \text{'}stay\text{'})$ is practically satisfied, e.g. the state trajectory is locally converged to s_c. To balance the efficiency and accuracy, The condition that s_c appears at least 8 times in the least 10 consecutive states is adopted for simulations in Section V.

$\boldsymbol{\omega} = [\omega_x, \omega_y, \omega_z]^T \in \mathbb{R}^3$ is the true angular velocity of $\mathcal{B}$ w.r.t. $\mathcal{N}$ expressed in $\mathcal{N}$, and

$$\boldsymbol{\Omega}(\boldsymbol{\omega}) = \begin{bmatrix} -\lfloor\boldsymbol{\omega}\times\rfloor & \boldsymbol{\omega} \\ -\boldsymbol{\omega}^T & 0 \end{bmatrix},\ \lfloor\boldsymbol{\omega}\times\rfloor = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}.$$

Second, $\boldsymbol{C}_N^B(\boldsymbol{q}) \in \mathbb{R}^{3\times3}$ is the rotation matrix from $\mathcal{N}$ to $\mathcal{B}$ expressed in $\mathcal{B}$ using quaternion elements [16], $\boldsymbol{C}_{\mathcal{N}}^{\mathcal{B}}(\boldsymbol{q}) =$

$$\begin{bmatrix} 2q_0^2 - 1 + 2q_1^2 & 2(q_1q_2 + q_0q_3) & 2(q_1q_3 - q_0q_2) \\ 2(q_1q_2 - q_0q_3) & 2q_0^2 - 1 + 2q_2^2 & 2(q_2q_3 + q_0q_1) \\ 2(q_1q_3 + q_0q_2) & 2(q_2q_3 - q_0q_1) & 2q_0^2 - 1 + 2q_3^2 \end{bmatrix}.$$

Then, the output of MARG sensors can be modeled[2] by

$$\begin{cases} \boldsymbol{\omega}^m = \boldsymbol{\omega} + \boldsymbol{b}^g + \boldsymbol{v}^g, \\ \boldsymbol{a} = \boldsymbol{C}_n^b(\boldsymbol{q})\boldsymbol{g} + \boldsymbol{b}^a + \boldsymbol{v}^a, \\ \boldsymbol{m} = \boldsymbol{C}_n^b(\boldsymbol{q})\boldsymbol{h} + \boldsymbol{b}^m + \boldsymbol{v}^m, \end{cases} \tag{11}$$

where $\boldsymbol{\omega}^m \in \mathbb{R}^3$ is the measured angular velocity, $\boldsymbol{m} \in \mathbb{R}^3$ is the measured Earth's magnetic field in $\mathcal{B}$ w.r.t. $\mathcal{N}$, $\boldsymbol{a} \in \mathbb{R}^3$ is the measured acceleration in $\mathcal{B}$ w.r.t. $\mathcal{N}$, $\boldsymbol{g} \in \mathbb{R}^3$ is the gravity vector, $\boldsymbol{h} \in \mathbb{R}^3$ is the Earth's magnetic field in $\mathcal{N}$, $\boldsymbol{b}^g, \boldsymbol{b}^a, \boldsymbol{b}^m \in \mathbb{R}^3$ are the bias of gyroscope, accelerometer and magnetometer, respectively, and $\boldsymbol{v}^g, \boldsymbol{v}^a, \boldsymbol{v}^m \in \mathbb{R}^3$ are assumed to be uncorrelated Gaussian noises of these sensors with zero mean and covariance matrices $\boldsymbol{\Sigma}_g = \sigma_g^2\boldsymbol{I}_3$, $\boldsymbol{\Sigma}_a = \sigma_a^2\boldsymbol{I}_3$ and $\boldsymbol{\Sigma}_m = \sigma_m^2\boldsymbol{I}_3$, where $\boldsymbol{I}_3$ denotes the identity matrix of dimension 3.

Next, we consider the state vector to be estimated as $\boldsymbol{x}_k = [\boldsymbol{q}_k; \boldsymbol{b}_k^g; \boldsymbol{b}_k^a; \boldsymbol{b}_k^m]$. After discretization, the state transition equation is formulated as $\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k) + \boldsymbol{w}_k = \text{blkdiag}(\boldsymbol{\Omega}_k, \boldsymbol{B}, \boldsymbol{I}_3, \boldsymbol{I}_3)[\boldsymbol{q}_k; \boldsymbol{b}_k^g; \boldsymbol{b}_k^a; \boldsymbol{b}_k^m] + [\boldsymbol{w}_k^q; \boldsymbol{w}_k^g; \boldsymbol{w}_k^a; \boldsymbol{w}_k^m]$, where $\boldsymbol{\Omega}_k = \boldsymbol{I}_3 + \frac{1}{2}\boldsymbol{\Omega}(\boldsymbol{\omega}_k^m - \boldsymbol{b}_k^g)T_e$ is the quaternion dynamic matrix [10], [17], and the gyroscope bias $\boldsymbol{b}_k^g$ is represented by a discrete Gaussian-Markov process, in which $\boldsymbol{B} = (1 - \beta^{-1})T_e\boldsymbol{I}_3$ is derived from $\dot{\boldsymbol{b}}_k^g = -\beta^{-1}\boldsymbol{b}_k^g + \boldsymbol{w}_k^g$ with β being the time constant of bias variation, and $T_e = t_{k+1} - t_k$ denotes the sampling interval, $\boldsymbol{b}_k^a$ and $\boldsymbol{b}_k^m$ are modeled as standard random walk, and $\boldsymbol{w}_k^q$, $\boldsymbol{w}_k^g$, $\boldsymbol{w}_k^a$ and $\boldsymbol{w}_k^m$ form the process noise, assumed to be uncorrelated Gaussian with zero mean and covariance matrices $\boldsymbol{\Sigma}_w^q = \sigma_{w,q}^2\boldsymbol{I}_3$, $\boldsymbol{\Sigma}_w^g = \sigma_{w,g}^2\boldsymbol{I}_3$, $\boldsymbol{\Sigma}_w^a = \sigma_{w,a}^2\boldsymbol{I}_3$ and $\boldsymbol{\Sigma}_w^m = \sigma_{w,m}^2\boldsymbol{I}_3$. As such, the process noise covariance matrix is diagonal and is represented by $\boldsymbol{Q} = \text{blkdiag}(\boldsymbol{\Sigma}_w^q, \boldsymbol{\Sigma}_w^g, \boldsymbol{\Sigma}_w^a, \boldsymbol{\Sigma}_w^m)$.

Subsequently, the measurement model is constructed by grouping measurements of accelerometer and magnetometer: $\boldsymbol{y}_k = [\boldsymbol{a}_k; \boldsymbol{m}_k] = g(\boldsymbol{x}_k) + \boldsymbol{v}_k = \text{blkdiag}(\boldsymbol{C}_n^b(\boldsymbol{q}_k), \boldsymbol{C}_n^b(\boldsymbol{q}_k))[\boldsymbol{g}; \boldsymbol{h}] + [\boldsymbol{v}_k^a; \boldsymbol{v}_k^m]$. Finally, the measurement noise covariance matrix is diagonal and is represented by $\boldsymbol{R} = \text{blkdiag}(\boldsymbol{\Sigma}_a^v, \boldsymbol{\Sigma}_m^v)$.

B. Monte Carlo numerical simulations

We use the ground truth of $\boldsymbol{b}^g$, $\boldsymbol{b}^a$ and q_{true} from Euroc database [15]. And q_{true} is used to solve out the $\boldsymbol{\omega}$. The

[2]In the simulation, we consider that the rigid body is just rotating on itself and its center of mass is not moving; thus the body linear acceleration expressed in $\mathcal{N}$ is zero.

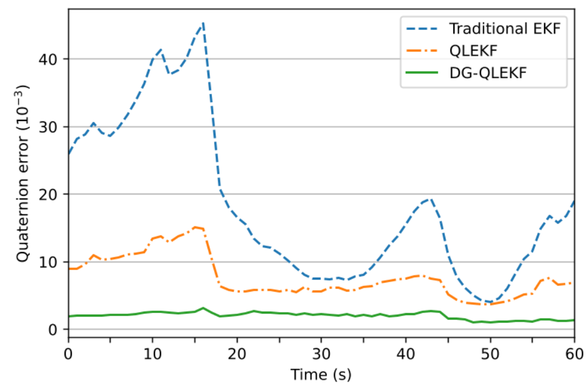


Fig. 2: Mean quaternion error (refer to Table I for the computation) among the traditional EKF, QLEKF, and DG-QLEKF after convergence of 50 Monte Carlo simulations

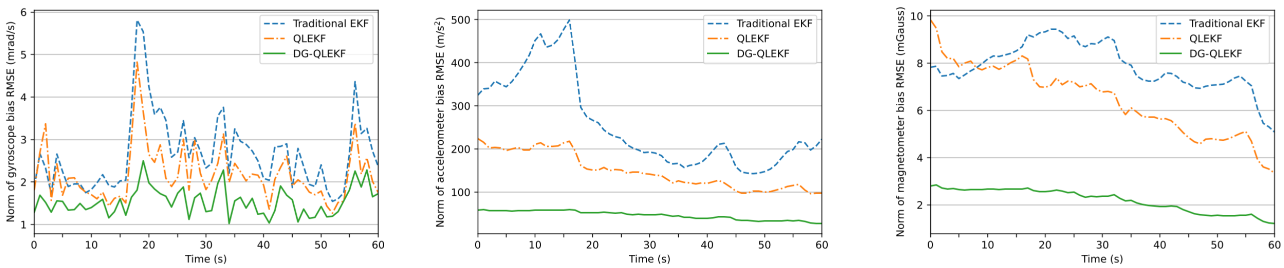
magnetometer bias ground truth is time-invariant as $\boldsymbol{b}^m = [5, 5, 5]$ mGauss. The sampling rate of the MARG sensors is set to 100 Hz, $\boldsymbol{g} = [0, 0, 9.81]$ m/s^2, $\boldsymbol{h} = [0.23, 0.01, 0.41]$ Gauss. For the initialization of each Monte Carlo simulation, we select $\boldsymbol{q}_0 = [0.5, 0.5, 0.5, 0.5]^T$, $\boldsymbol{b}_0^g = [2.2, 2, 2]$ mrad/s, $\boldsymbol{b}_0^a = [0.1, 0.1, 0.1]$ m/s^2, $\boldsymbol{b}_0^m = [0.1, 0.1, 0.1]$ Gauss, $\boldsymbol{P}_0 = 10\boldsymbol{I}_{13}$, r and q take values sequentially from the list $\{2, 4, 8\}$, $\underline{r} = \underline{q} = 10^{-3}$, $\overline{r} = \overline{q} = 10^3$, $n_{ratio} = 5$, $\sigma_{w,q} = 10^{-4}$, $\sigma_{w,g} = 10^{-3}$ rad/s, $\sigma_{w,a} = 10^{-2}$ m/s^2, $\sigma_{w,m} = 10^{-4}$ Gauss, $\sigma_{v,a} = 2 \times 10^{-2}$ m/s^2, $\sigma_{v,m} = 2 \times 10^{-3}$ Gauss, $\beta = 100$, $\boldsymbol{Q}_0 = \text{blkdiag}(10^{-8}\boldsymbol{I}_4, 10^{-6}\boldsymbol{I}_3, 10^{-4}\boldsymbol{I}_3, 10^{-8}\boldsymbol{I}_3)$, $\boldsymbol{R}_0 = \text{blkdiag}(4 \times 10^{-4}\boldsymbol{I}_3, 4 \times 10^{-6}\boldsymbol{I}_3)$, and $(\boldsymbol{Q}_0, \boldsymbol{R}_0)$ is used in the traditional EKF. In terms of QLEKF, $M = N = 5$, $\{\boldsymbol{Q}_k^{(i)}\}$ and $\{\boldsymbol{R}_k^{(j)}\}$ are set as a geometric progression with a ratio of 10 and $\boldsymbol{Q}^{(3)} = \boldsymbol{Q}_0$, $\boldsymbol{R}^{(3)} = \boldsymbol{R}_0$. The Q-learning search starts at the grid $(\boldsymbol{Q}^{(2)}, \boldsymbol{R}^{(2)})$. For each Monte Carlo simulation, we run 360 periods and 100 time steps for each iteration. The learning rate, discount factor and random action selection probability are fixed to $\alpha = 0.1$, $\gamma = 0.9$ and $\epsilon = 0.1$, respectively.

Table I shows that after the convergence of 50 Monte Carlo simulations, the three EKFs deliver different state estimation performances. Generally, the QLEKF outperforms the traditional EKF, particularly improving the quaternion error by 58.34%. As for the DG-QLEKF, it exhibits a prominent predominance of all statistics compared to both the traditional EKF and QLEKF. In detail, the average improvements in quaternion error, RMSEs of gyroscope bias, accelerometer bias, and magnetometer bias of DG-QLEKF compared to the traditional EKF are 88.93%, 40.60%, 82.39% and 74.62%, while those improvements of DG-QLEKF compared to QLEKF are 73.42%, 29.70%, 75.21%, and 69.22%, respectively. As shown in Fig. 2, the three curves share similar tendencies and fluctuations after convergence. In contrast, the quaternion error of DG-QLEKF is always distinctly lower than the other two EKFs in the whole convergence period. Similar behaviors can be observed from Fig. 3, but with more unstable variations of QLEKF.

TABLE I: Mean of 50 Monte Carlo simulations after convergence for quaternion error and Root Mean Square Error (RMSE) of bias of gyroscope, accelerometer, and magnetometer

	Mean of quaternion error* ($\times 10^{-3}$)	RMSE of gyroscope bias (mrad/s)			RMSE of accelerometer bias (m/s^2)			RMSE of magnetometer bias (mGauss)		
		x-axis	y-axis	z-axis	x-axis	y-axis	z-axis	x-axis	y-axis	z-axis
Traditional EKF	18.35	1.564	1.534	1.184	0.135	0.183	0.064	6.756	1.744	3.580
QLEKF	7.644	1.231	1.116	1.133	0.120	0.067	0.048	4.942	1.089	3.963
DG-QLEKF	**2.032**	**0.760**	**0.643**	**1.038**	**0.043**	**0.014**	**0.008**	**1.816**	**0.277**	**1.196**

* For a ground truth unit quaternion $\boldsymbol{q}_{true}$, the quaternion error of its unit estimate $\hat{\boldsymbol{q}}$ is computed as $\|\boldsymbol{q}_{true}^{-1} \otimes \hat{\boldsymbol{q}} - \boldsymbol{q}_I\|$, where $\boldsymbol{q}_I = [1\ 0\ 0\ 0]^T$ is the identity quaternion.



(a) Norm of gyroscope bias RMSE (b) Norm of accelerometer bias RMSE (c) Norm of magnetometer bias RMSE

Fig. 3: Norm of RMSE of gyroscope bias, accelerometer bias, and magnetometer bias among the traditional EKF, QLEKF, and DG-QLEKF after convergence of 50 Monte Carlo simulations

VI. CONCLUSIONS

In this paper, the Q-learning-based EKF (QLEKF) for state estimation of nonlinear systems was introduced as a premise to address the often-cumbersome tuning of noise covariance matrices in the EKF. To overcome the existing drawbacks of QLEKF resulting from heuristics in designing Q-learning, a dynamic grid-based Q-learning EKF algorithm (DG-QLEKF) has been proposed, which can thoroughly exploit an arbitrary search scope to find appropriate values of noise covariance matrices, leading to excellent state estimation. Through Monte Carlo numerical simulations based on real flight data from an unmanned aerial vehicle, the DG-QLEKF, on average, has revealed much more improvement in attitude and biases estimation after convergence compared to the traditional EKF and QLEKF. Future work on the DG-QLEKF can be undertaken by adapting inter-process and inter-measurement noises covariance matrices.

REFERENCES

[1] G. Welch, "An introduction to the Kalman filter," *SIGGRAPH Tutorial*, 2001.

[2] R. G. Brown and P. Y. C. Hwang, *Introduction to random signals and applied Kalman filtering : With MATLAB exercises and solutions*. New York, John Wiley & Sons, Inc., 1997.

[3] M. Zmitri, H. Fourati, and C. Prieur, "BiLSTM network-based extended kalman filter for magnetic field gradient aided indoor navigation," *IEEE Sensors Journal*, vol. 22, no. 6, pp. 4781–4789, 2022.

[4] J. L. Crassidis, F. L. Markley, and Y. Cheng, "Survey of nonlinear attitude estimation methods," *Journal of guidance, control, and dynamics*, vol. 30, no. 1, pp. 12–28, 2007.

[5] F. Auger, M. Hilairet, J. M. Guerrero, E. Monmasson, T. Orlowska-Kowalska, and S. Katsura, "Industrial applications of the Kalman filter: A review," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 12, pp. 5458–5471, 2013.

[6] C. J. Watkins, "Learning from delayed rewards," *PhD thesis, Cambridge University*, 1989.

[7] A. Maoudj and A. L. Christensen, "Q-learning-based navigation for mobile robots in continuous and dynamic environments," in *2021 IEEE 17th International Conference on Automation Science and Engineering (CASE)*, pp. 1338–1345, 2021.

[8] K. Xiong, C. Wei, and H. Zhang, "Q-learning for noise covariance adaptation in extended KALMAN filter," *Asian Journal of Control*, vol. 23, no. 4, pp. 1803–1816, 2021.

[9] C. J. Watkins and P. Dayan, "Q-learning," *Machine learning*, vol. 8, no. 3-4, pp. 279–292, 1992.

[10] X. Dai, V. Nateghi, H. Fourati, and C. Prieur, "Q-learning-based noise covariance adaptation in Kalman filter for MARG sensors attitude estimation," in *IEEE International Symposium on Inertial Sensors & Systems*, Avignon, France, May 2022. Accepted version: https://hal.laas.fr/INRIA/hal-03555546v1.

[11] R. S. Sutton and A. G. Barto, "Reinforcement learning: An introduction," *MIT press*, 2018.

[12] S. Thrun, "Efficient exploration in reinforcement learning," *Technical Report. Carnegie Mellon University*, 1992.

[13] S. D. Whitehead, "A complexity analysis of cooperative mechanisms in reinforcement learning," in *Proceedings of the ninth National Conference on Artificial Intelligence-Volume 2*, pp. 607–613, 1991.

[14] J. Bridle, "Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters," *Advances in neural information processing systems*, vol. 2, 1989.

[15] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, "The EuRoC micro aerial vehicle datasets," *The International Journal of Robotics Research*, vol. 35, no. 10, pp. 1157–1163, 2016.

[16] J. B. Kuipers, *Quaternions and Rotation Sequences: A Primer with Applications to Orbits, Aerospace, and Virtual Reality*. Princeton University press, 1999.

[17] J. R. Wertz, *Spacecraft Attitude Determination and Control*, vol. 73. Springer Science & Business Media, 2012.