**ORIGINAL ARTICLE**

# A low-cost integrated navigation system based on factor graph nonlinear optimization for autonomous flight

Sina Taghizadeh[1] · Mohsen Nezhadshahbodaghi[2] · Reza Safabakhsh[1] · Mohammad Reza Mosavi[2]

## Abstract

We propose a low-cost integrated navigation system to enhance navigation accuracy using multiple sensors such as the inertial measurement unit (IMU) and Global Positioning System (GPS). Due to the heterogeneity of the employed sensors, the suggested navigation system is equipped with a graph-based optimizer formulated as a maximum a posteriori estimator. As already known, graphic optimization methods are computationally more complex. To reduce this complexity, high-frequency IMU measurements are pre-integrated between lower-frequency GPS measurements. These pre-integrated measurements can be utilized in conjunction with the magnetometer and barometer sensor measurements to carry out navigation in case of GPS weak signal reception. The resulting optimization graph will be solved using the marginalization technique to attain a less computationally intensive optimization favorable in real-time applications. Incorporating internal sensors in navigation will come with problems like the initialization phase. This problem is tackled by integrating the inertial measurements between the required states in the body frame of the last state and then transforming them into the world frame. To evaluate the applicability of the proposed method in real-time situations, a dataset collected by a SkywalkerX8 Unmanned Aerial Vehicle is utilized. This dataset includes all the necessary maneuvers, including climb, descent, turn and cruise during the flight time. The results confirm that our proposed integrated system has superior statistical measures and temporal performance compared to the extended Kalman filter (EKF); it outperforms the conventional Factor Graph Optimization and EKF by approximately 45%.

**Keywords** Integration navigation · Factor graph · Optimization

## Introduction

Autonomous vehicles have been widely developed for civilian service, industrial and military applications. Thanks to advances in navigation sensors and global navigation satellite systems (GNSS), a large percentage of unmanned aerial

vehicle (UAV) missions can be performed autonomously. One of the most critical tasks in UAV missions is the landing phase. During the landing phase of UAVs, severe GNSS problems can occur, such as interference, lousy weather conditions, sensor faults and poor runway alignment. The landing phase is often performed manually to avoid such failures but is prone to human error. Consequently, it is necessary to find a method that provides accurate information to the autopilot for automated landing.

Accurate measurements regarding UAV's location and speed can be obtained by different instrumentation solutions, such as Instrument Landing System (ILS) (Eltahier and Hamid 2017), radar and laser-based methods, which require expensive infrastructures. One can utilize other light and inexpensive navigation sensors such as IMU, magnetometer, and GNSS to estimate the vehicle's state by exploiting various sensor integration schemes (Groves 2015). Indeed, integrated navigation systems are the best solutions to

✉ Mohammad Reza Mosavi
  M_Mosavi@iust.ac.ir

  Sina Taghizadeh
  R.Taghizadeh@aut.ac.ir

  Mohsen Nezhadshahbodaghi
  m_nezhadshahbodaghi@elec.iust.ac.ir

  Reza Safabakhsh
  safa@aut.ac.ir

[1] Department of Computer Engineering, Amirkabir University of Technology, Tehran 15875-4413, Iran

[2] Department of Electrical Engineering, Iran University of Science and Technology, Tehran 16846-13114, Iran

compensate for limitations and achieve automatic navigation while balancing cost and precision.

There are various approaches for heterogeneous sensor integration. These methods are classified into two groups. The first group includes the filter-based methods that consider the last state of the integrated system (Nezhadshahbodaghi et al. 2021). An essential issue of filter-based algorithms is that the number of states restricts their performance. As the number of states grows, the computing complexity of such algorithms will increase extensively (Rogers 2007). The second group of integration methods consists of smoothing-based approaches, also known as the optimization-based approaches. These methods estimate the entire history of the states by solving a large number of linear algebraic equations (Carlone et al. 2014).

A smoothing-based approach is chosen in this study since it is more accurate than filter-based methods (Foster et al. 2016). Also, a sliding window scheme is employed to avoid processing a growing number of inertial measurements and system states (Kaess et al. 2012). The graphical factor method is chosen to solve the navigation problem. This method enables the algorithm to be carried out in parallel and overcomes the sparseness of the sensor measurements (Kschischang et al. 2001).

To solve the problem of changing state estimation in every step of the optimization, it is necessary to integrate the inertial observations between two consecutive states, with motion considered a parameter. To verify the performance of the suggested integration, we used a dataset collected by a UAV equipped with the IMU, GNSS receiver, magnetometer and barometer. The results demonstrate that the proposed system has a significant improvement in terms of the estimations of location and velocity of the vehicle compared to the traditional EKF. We should mention that our proposed model has many new contributions:

- Applying the pre-integration technique to the INS measurements reduces the number of measurement nodes of the factor graph.
- Fixed timestep formulation of vehicle's motion.
- Solving the navigation factor graph of a fixed-wing UAV by marginalizing the prior measurements; thus, acquiring a smaller optimization problem that can be solved faster.
- Validating our method by the collected flight dataset of the SkywalkerX8 UAV.

The next section discusses and reviews various smoothing and filter-based algorithms and their significant contributions to each approach. The integrated navigation section describes the advantages and disadvantages of the multisensor navigation solution concerning autonomous landing. The implementation procedure of the proposed algorithm will be explained from the theoretical and innovation perspective in the methodology section. The remaining two sections discuss the achieved temporal and statistical results and conclude this study by comparing our method to FGO and EKF methods.

## Related work

Existing navigation methods can be categorized into active and passive approaches. The active methods use extrinsic signals such as GNSS signals. These methods usually suffer from noise and GNSS-denied environments. As a result, navigation algorithms should be integrated with other passive sensors such as IMU and cameras. The second group of navigation approaches is passive or self-contained navigation. In this group, the states of a vehicle are estimated from various sensor measurements, such as magnetic and barometric measurements (Nezhadshahbodaghi and Mosavi 2021). Lynen et al. (2013) proposed a navigation system in which the EKF integrates the measurements from a theoretically unlimited number of different sensors. Lynen's method can calibrate heterogeneous sensors and compensate for time-delayed measurements. Unscented Kalman filter (UKF)-based vision/laser/GPS and cubature Kalman filter (CKF)-based integration systems were presented by Shen et al. (2014) and Nourmohammadi and Keighobadi (2017), respectively. Shen's UKF method has two main innovations: using an augmented state vector that has been formed from the current and past system states and applying UKF to integrate the heterogeneous sensors. Nourmohammadi and Keighobadi's algorithm improved navigation accuracy with a CKF-based decentralized integration algorithm. Their method utilized two independent filters to estimate position, velocity and orientation to avoid position and velocity errors propagation through the orientation estimation filter.

A multistate constraint Kalman filter was also introduced to provide a new measurement model for the Kalman filter, estimating and updating the states of the system (Mourikis and Roumeliotis 2007; Jones and Soatto 2011). Lubton and Sukkarieh (2012) presented a new method to integrate IMU and visual sensors. In their method, the initial values of the inertial sensors can be adequately provided by assuming constant motion acceleration in each step, resulting in a linear evolution of the moving body's velocity and integrating the inertial sensor values in the body-fixed frame. Since the above methods utilize filter-based structures to integrate the information of the different sensors, they are usually restricted by the number of the system states. Indelman et al. (2013) utilized a new IMU/vision integration system via a factor graph in the class of graph-based integration methods. Their integration made adding additional sensors as simple as plug-and-play, requiring no specific algorithmic corrections. Artal and Tardos (2017) developed a tightly-coupled

**Fig. 1** Important elements in the landing phase. The autonomous landing phase requires additional measurements, such as the $\gamma$ angle, in contrast to the cruising phase, which does not require such information
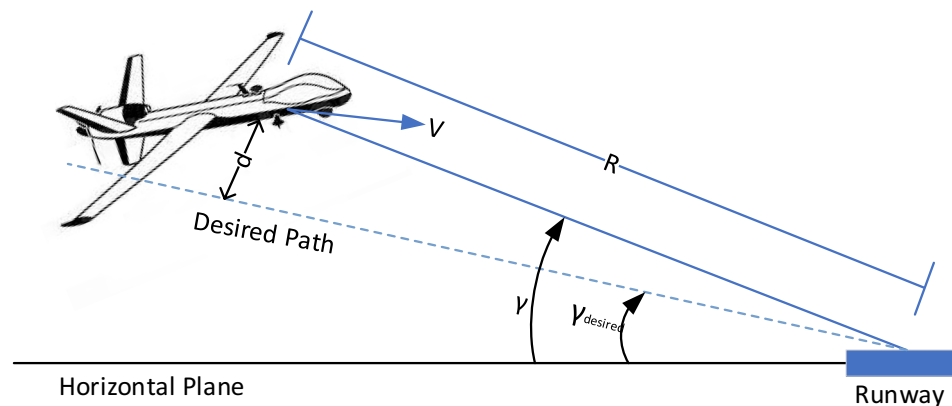


visual-inertial Simultaneous Localization and Mapping (SLAM) system using the graphical integration idea and monocular camera and offered a new method of IMU initialization. Mascaro et al. (2018) proposed the separation of state estimation and sensor integration to achieve online navigation in real-time. A tightly-coupled Maximum a Posteriori (MAP) probability method solved by factor graphs was also used by Li et al. (2018). They aimed to provide accurate localization of a ground vehicle in urban areas.

Many traditional GNSS/INS integration methods use only GNSS signals (Wen et al. 2020; Gu et al. 2019), whereas one can combine several measurements such as GPS, barometers, and magnetometers to achieve superior performance in GNSS-denied environments. As Jizhou et al. (2019) and Dai et al. (2020) have proposed, multi-sensor integration systems can be used to increase positioning accuracy in GNSS-deficient environments. These integrated systems require high computation complexity and are not suitable for real-time applications. Wei et al. (2021) and Xu et al. (2021) have also presented Visual Odometry (VO)/GPS/INS integration systems based on factor graphs. Although they could achieve a good performance, the computational complexity of the VO algorithm restricted its usage in real-time applications. This algorithm was also unstable in GNSS-deficient environments. Our proposed method compensates for this weakness by using multiple sensors. More optimization-based integration methods of the passive and active sensors can be found in Qin (2019).

## Integrated navigation

Integrated navigation can significantly improve the vehicle's position, velocity and orientation estimation. This section will describe the integration method and advantages of using auxiliary sensors to make a sound navigation system.

## Multi-sensor navigation advantages

As depicted in Fig. 1, accurate longitudinal and lateral control of the UAV in the landing phase is required in addition to standard navigational information. These pieces of information can be achieved by considering the 3D position, vertical speed and the gamma angle $\gamma = \sin^{-1}\left(\frac{\hat{v}}{\hat{v}_d}\right)$ of the vehicle.

Position, velocity and attitude errors derived from an inertial navigation system suffer from sensor drifts. The main reason for these drifts is uncertainty in the calibration parameters of the sensors, e.g., bias and scale. Other measurement systems such as GNSS that provide independent navigation information can overcome these errors by resetting measurements (Titterton and Weston 2004). GNSS signals can encounter severe signal blockages or weak receptions. To aid GNSS in signal outage areas, most current systems employ heterogeneous multi-sensors such as magnetometers and air data sensors (Groves 2015). As Fig. 2 shows, in a filter-based integration navigation system, the errors of the auxiliary sensors are estimated to enhance the system's accuracy (Rogers 2007).

## Optimization-based integration

Because of biased and noisy measurements of different navigation sensors, state estimation has been widely used to achieve a reliable navigation system (Lynen et al. 2013). Integrating the outputs of heterogeneous sensors, also known as multiple sensor integration, is a means to aid the system's navigation task by providing a reliable and robust description of the system's states (Shen et al. 2014). The EKF method can achieve a magnificent performance for the linearized model of nonlinear dynamics with Gaussian noise. However, it will struggle with non-Gaussian noises. In systems with significant uncertainties, such as nonlinear systems with non-Gaussian noise, because of considering such uncertainties as a Gaussian noise process,

**Fig. 2** Block diagram of an integrated navigation system. Readings from multiple sensors are compared to INS readings, and computed errors relative to the INS data are fed through the designed filter and are fused to obtain a better overall error estimation. Finally, state evaluation is carried out by adding the filter error estimation to the INS readings. Once the final output of the filter is calculated, it can be fed back to the INS mechanization block for the next round of filter computation
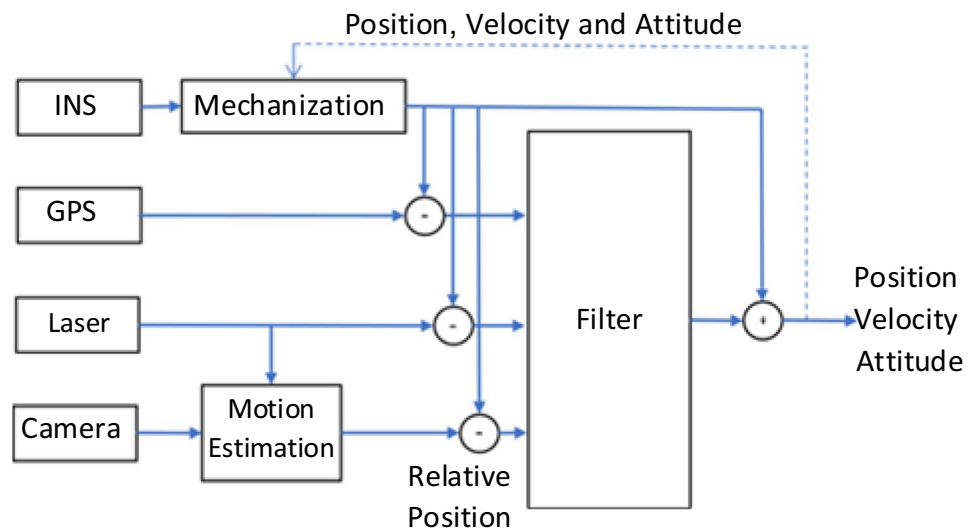


the EKF estimator will suffer from significant estimation errors (Groves. 2015). To overcome this problem, uncertainty can be handled as a state variable in the system dynamic model (Rogers 2007). Indeed, there are several GNSS/INS integration issues, such as large uncertainties because of low-cost and light microelectromechanical sensors. Developing a nonlinear method based on a factor graph reduces the uncertainty effect and replaces the EKF in integrated navigation systems (Kschischang et al. 2001). In this development, the integration problem is formulated in a MAP. From a Bayes theorem standpoint, the MAP improves navigation states by using nonlinear optimization (Kaess et al. 2012).

## Methodology

Obtaining and integrating measurements from different heterogeneous sensors with nonlinear outputs is a computationally complex task. This study proposes a new graph-based method to integrate multiple heterogeneous sensors for real-time applications.

### Factor graph

A factor graph is a graphical model that encodes the conditional probability value between the measured and unknown variables (Kschischang et al. 2001). Suppose that the function $g(x_1, x_2, \ldots, x_n)$ is expressible as products of local functions. Every one of these local functions will have a domain which is a subset of set $\{x_1, x_2, \ldots, x_n\}$. So, the general function can be written as follows:

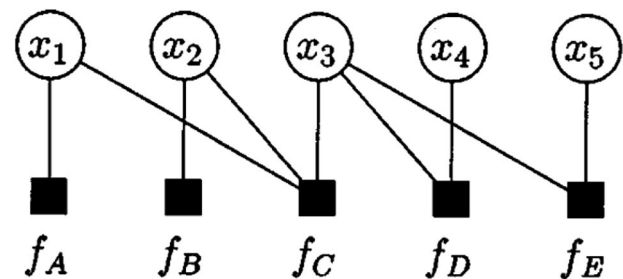$$g(x_1, x_2, \ldots, x_n) = \prod_{j \in J} f_j(X_j) \tag{1}$$



**Fig. 3** Factor graph for the multiplication of $f_A(x_1)f_B(x_2)f_C(x_1,x_2,x_3)f_D(x_3,x_4)f_E(x_3,x_5)$. As shown in the function is factored as a multiplication of five functions with different domains, all subsets of the $\{x_1, x_2, \ldots, x_5\}$

where $J$ is the number of local functions. $X_j$ denotes the j-th subset of $\{x_1, x_2, \ldots, x_n\}$ and $f_j(X_j)$ represents the function of the values $X_j$. The graph factor is a two-dimensional graph that provides the factorization structure, as shown in Eq. (1). In this graph, there are two kinds of nodes, called variable and function nodes, so that if $x_i$ is a member of the domain of $f_j$, an edge is drawn between them. For example, $g(x_1, x_2, x_3, x_4, x_5)$ can be given as follows:

$$g(x_1, x_2, x_3, x_4, x_5) = f_A(x_1)f_B(x_2)f_C(x_1, x_2, x_3)f_D(x_3, x_4)f_E(x_3, x_5) \tag{2}$$

$g$ is described as a multiplication of five factors. According to (1), $J = \{A, B, C, D, E\}$, $X_A = \{x_1\}$, $X_B = \{x_2\}$, $X_C = \{x_1, x_2, x_3\}$, $X_D = \{x_3, x_4\}$, and $X_E = \{x_3, x_5\}$. The graph factor of (2) is shown in Fig. 3 for clarification.

### Integrating sensors by factor graph

By choosing a factor graph structure for the proposed navigation system, the graph-based optimization algorithm is

applied to solve the nonlinear problem and predict the current state with the information of multiple heterogeneous sensors.

The system state vector at the time $t_i$ is represented by the IMU measurements of the orientation, position and velocity given by Eq. (3) (Forster et al. 2016).

$$x_i = [R_i, p_i, v_i, b_i] \tag{3}$$

where $R_i$ is a three-by-three orientation matrix. Variables $p_i$ and $v_i$ denote the position and velocity of the UAV at the time $t_i$, respectively. Furthermore, $b_i = [b_i^g, b_i^a]$ shows the biases of the inertial sensors, e.g., gyroscope and accelerometer. It is worth noting that the state $(R_i, p_i)$, $v_i$, and $b_i$ belong to $SE(3)$, $R^3$ and $R^6$, respectively.

The UAV state variable at the time $t_k$ can be considered as $X_k = \{x_i\}_{i=0}^k$. According to the primary statistical distribution of $p(X_0)$, $X_k$. with the probability distribution of $p(X_k|X_{k+1})$ is propagated through the vehicle dynamics. Also, $Z_k \in R^m$ denotes the independent measurements at the time $t_k$. The Bayes estimation consists of two steps called prediction and update. The prediction step is carried out by considering the prior probability distribution of the state model, and in the update step, the goal is correct the prior distribution using the newest measurements to obtain the posterior probability distribution of $p(X_k|Z_k)$. According to the measurements available up to the current time $t_k$, the conditional probability distribution function of $p(X_k|Z_k)$ for the state vector of $X_k$ is a marginalized function. The Bayes rule allows for factor $p(X_k|Z_k)$ as follows (Forster et al. 2016):
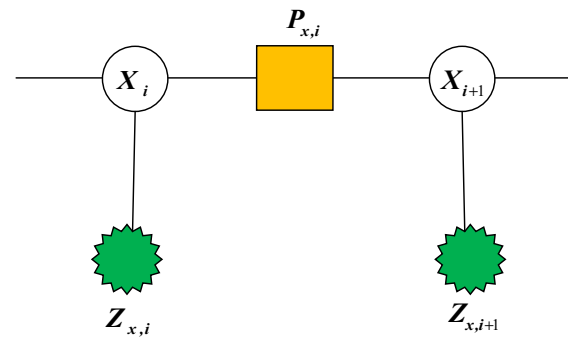


**Fig. 4** Factor graph of the state ($X$) and sensor measurement ($Z$). The nodes are made of either factors or states, and an edge can exist between factor nodes and state nodes. Each factor node represents a norm function between the measurement and its prediction. In this picture, factor nodes are represented by colored boxes and states as circles, $P$ represents the state factor node, and $Z$ is the measurement factor node

As depicted in Fig. 4, the factor graph model of the proposed navigation system is specified by the local functions of the factor nodes of $P_{x,i}$ and $Z_{x,i}$, given by $f(X_i|X_{i-1})$ and $f(Z_i|X_i)$, respectively (Indelman et al. 2013). For more explanation, the factor graph model of an IMU/GPS integrated navigation system can be designed, as shown in Fig. 5.

## Factor graph optimization

Since the sampling frequencies of the heterogeneous sensors are different, the factor graph model can describe the

$$
\begin{aligned}
p(X_k|Z_k) &= \frac{p(Z_k|X_k).p(X_k|X_{k-1})}{p(Z_k|Z_{k-1})} \cdot p(X_{k-1}|Z_{k-1}) \\
&= \frac{p(Z_k|X_k).p(X_k|X_{k-1})}{p(Z_k|Z_{k-1})} \cdot \frac{p(Z_{k-1}|X_{k-1}).p(X_{k-1}|X_{k-2})}{p(Z_{k-1}|Z_{k-2})} \cdot p(X_{k-2}|Z_{k-2}) \\
&= \frac{p(Z_k|X_k).p(X_k|X_{k-1})}{p(Z_k|Z_{k-1})} \cdot \frac{p(Z_{k-1}|X_{k-1}).p(X_{k-1}|X_{k-2})}{p(Z_{k-1}|Z_{k-2})} \dots \cdot \frac{p(Z_1|X_1).p(X_1|X_0)}{p(Z_0)} \cdot p(X_0) \\
&= \prod_{i=1}^{k} \frac{p(Z_i|X_i).p(X_i|X_{i-1})}{p(Z_i|Z_{i-1})} \cdot p(X_0)
\end{aligned}
\tag{4}
$$

$p(X_0)$ represents all available prior information. The main concern in navigation problems is acquiring the current optimal navigation solution. By factorizing the probability of $p(X_k|Z_k)$ given by Eq. (4) State with the MAP estimate can be obtained as follows (Forster et al. 2016):

$$\hat{X}_i^{MAP} = \arg\max_{X_i} \left(-\log\left(p(X_i|Z_i)\right)\right) \tag{5}$$

characteristics of such a navigation system appropriately. Our graph-based integration structure algorithm is presented in Fig. 6. The proposed model can handle the situations in which the heterogeneous sensors provide the measurements in different intervals. Factor graphs are more convenient in adding nonlinear sensor models and can easily remove an unavailable sensor from the optimization procedure by halting the addition of its upcoming associated measurement nods.
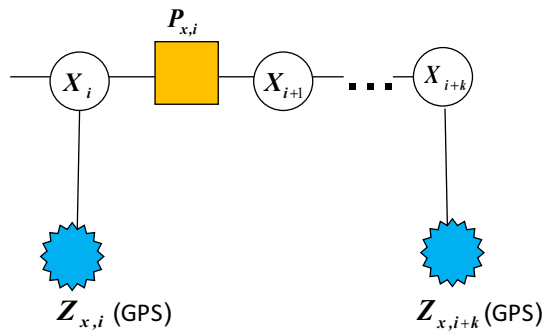
**Fig. 5** Factor graph of the IMU/GPS integrated navigation system. One of the advantages of using factor graphs to solve navigation systems is that these algorithms can solve navigation problems dealing with unsynchronized measurements such as integrated high-frequency IMU and low-frequency GPS sensors. All IMU measurements between two consecutive GPS measurements are pre-integrated to a single node to attain a less computationally intensive factor graph problem

The proposed graph-based integrated navigation system with heterogeneous sensors is implemented through the following seven steps.

### Step 1: IMU model and pre-integration

The inertial sensors provide the angular velocities and specific forces of the UAV by a 3-axis gyroscope and accelerometer that are rigidly attached to the UAV body so that their measurements can be formulated as follows (Titterton and Weston 2004):

$$\omega_{WB}'^{B}(t) = \omega_{WB}^{B}(t) + b^{g}(t) + \eta^{g}(t) \tag{6}$$

$$a^{B}(t) = R_{WB}^{T}(t)\left(a^{W}(t) - g^{W}(t)\right) + b^{a}(t) + \eta^{a}(t) \tag{7}$$

where $\omega_{WB}'^{B}(t)$ and $\omega_{WB}^{B}(t)$ represent the estimated and actual measurements of the angular velocity in the body frame (B), respectively. $a^{B}(t)$ and $a^{W}(t)$ are the linear accelerations of the UAV in the body (B) and the world (W) frames, respectively. $b(t) = [b^{g}(t), b^{a}(t)]$ denotes the gyroscope and accelerometer sensors' biases. $\eta(t) = [\eta^{g}(t), \eta^{a}(t)]$ denotes the additive white noises of the inertial sensors. Also, $g^{W}(t)$ is the gravity vector in the world frame. Finally, $R_{WB}(t)$ is utilized to transform a vector or matrix from the body frame to the world frame. It is noteworthy that we neglected the Coriolis acceleration.

The continuous IMU dynamic model of the presented structure is as follows (Titterton and Weston 2004):

$$\dot{p}^{W}(t) = v^{W}(t) \tag{8}$$

$$\dot{v}^{W}(t) = a^{W}(t) \tag{9}$$

$$\dot{R}_{WB}(t) = R_{WB}(t)\left[\omega_{WB}^{B}(t)\right]\times \tag{10}$$

$$\dot{b}^{g}(t) = 0 \tag{11}$$

$$\dot{b}^{a}(t) = 0 \tag{12}$$

where $p^{W}(t)$ and $v^{W}(t)$ are the position and velocity of the UAV in the world frame. $\left[\omega_{WB}^{B}(t)\right]\times$ denotes the skew-symmetric matrix that can be described as follows:

$$\left[\omega_{WB}^{B}(t)\right]\times = \begin{bmatrix} 0 & -\omega_{z} & \omega_{y} \\ \omega_{z} & 0 & -\omega_{x} \\ -\omega_{y} & \omega_{x} & 0 \end{bmatrix} \tag{13}$$

The conventional initialization of inertial sensors suffers from several problems. Since all inertial measurements are sampled at very high rates and transformed into the world frame before integration, their updates in the EKF filter must be performed at very high rates. Another problem is that using the batch-based filters is restricted by the traditional initialization (Lupton and Sukkarieh 2012). We tackled these problems by integrating the inertial measurements between the required states in the body frame of the last state and then transformed them into the world frame. For this purpose, we have rewritten (8) to (10) to evaluate the state of the UAV at time $t + \Delta t$ as:

$$p^{w}(t + \Delta t) = p^{w}(t) + \int_{t}^{t+\Delta t} v^{W}(\tau)d\tau + \iint_{t}^{t+\Delta t} a^{W}(\tau)d\tau^{2} \tag{14}$$

$$v^{W}(t + \Delta t) = v^{W}(t) + \int_{t}^{t+\Delta t} a^{W}(\tau)d\tau \tag{15}$$

$$R_{WB}(t + \Delta t) = R_{WB}(t)Exp\left(\int_{t}^{t+\Delta t} \omega_{WB}^{B}(\tau)d\tau\right) \tag{16}$$

By approximating $\omega_{WB}^{B}(t)$ and $a^{W}(t)$ being constant in the time interval $[t, t + \Delta t]$, the above equations can be rewritten as follows:

$$p^{w}(t + \Delta t) = p^{w}(t) + v^{W}(t)\Delta t + \frac{1}{2}a^{W}(t)\Delta t^{2} \tag{17}$$

$$v^{W}(t + \Delta t) = v^{W}(t) + a^{W}(t)\Delta t \tag{18}$$

$$R_{WB}(t + \Delta t) = R_{WB}(t)Exp\left(\omega_{WB}^{B}(t)\Delta t\right) \tag{19}$$

By using (6) and (7), $\omega_{WB}^{B}(t)$ and $a^{W}(t)$ can be rewritten as a function of the IMU measurements given by Eqs. (20) to (22).

**Fig. 6** Graph-based integrated navigation structure. Various sensors are integrated to form a complex navigation system. In general, systems with this variety of sensors always will experience desynced data due to variation in computation times or algorithms task scheduling even if sensor frequencies are forced to be in sync
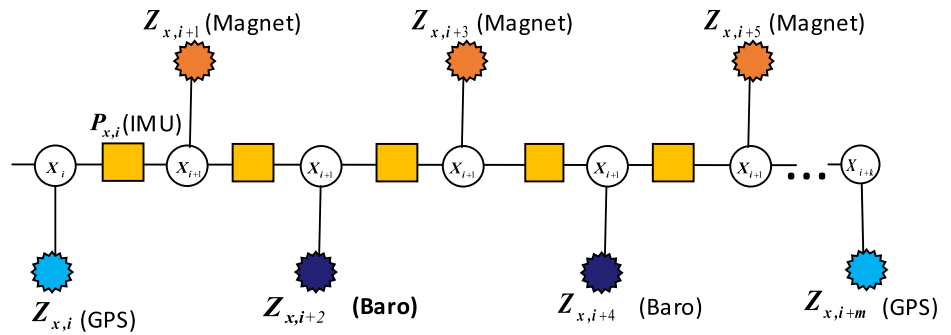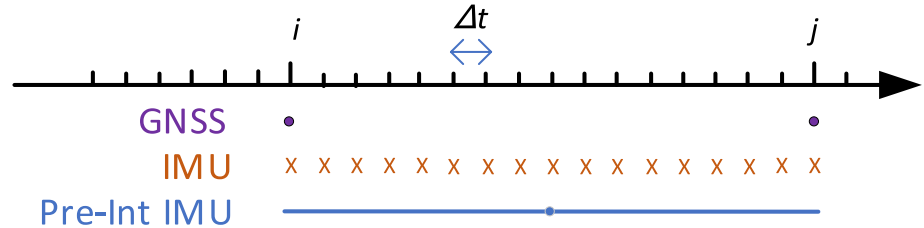


**Fig. 7** Difference sampling frequencies of IMU and GNSS. The high-frequency IMU measurements are pre-integrated between lower-frequency GNSS measurements to decrease the resultant size of the factor graph



$$p(t + \Delta t) = p(t) + v(t)\Delta t + \tfrac{1}{2}g\Delta t^2 + \tfrac{1}{2}R(t)\big(a(t) - b^a(t) - \eta^{ad}(t)\big)\Delta t^2 \tag{20}$$

$$v(t + \Delta t) = v(t) + g\Delta t + R(t)\big(a(t) - b^a(t) - \eta^{ad}(t)\big)\Delta t \tag{21}$$

$$R(t + \Delta t) = R(t)Exp\big(\big(\omega'(t) - b^g(t) - \eta^{gd}(t)\big)\Delta t\big) \tag{22}$$

$\eta^{ad}(t)$ and $\eta^{gd}(t)$ denote the covariance of the additive white noises of the accelerometer and gyroscope, respectively. In the above equations, frame subscripts are dropped for convenience. The value of $R(t)$ is assumed constant during two consecutive measurements. According to (10), this assumption is incorrect in practical applications and can be fulfilled by utilizing a high-rate IMU (Forster et al. 2016). Since (20) to (22) relate states at time $t$ to the states at time $t + \Delta t$, the high measurement frequency of the IMU sensor will result in a massive amount of measurement nodes of the factor graph approach. To circumnavigate this problem, IMU measurements are pre-integrated between two consecutive measurements of the lower frequency GNSS sensor (Mu et al. 2018). Figure 7 depicts the pre-integration and synchronizing procedure of the IMU and GNSS sensors.

As Fig. 7 represents, integrating all of the IMU observations between two sequential GNSS measurements at the timesteps $k = i$ and $k = j$, Eqs. (23) to (25) can be derived.

$$R_j = R_i \prod_{k=i}^{j-1} exp\big(\big(\omega'_k - b^g_k - \eta^{gd}_k\big)\Delta t\big) \tag{23}$$

$$v_j = v_i + g\Delta t_{ij} + \sum_{k=i}^{j-1} R_k\big(a_k - b^a_k - \eta^{ad}_k\big)\Delta t \tag{24}$$

$$p_j = p_i + \sum_{k=i}^{j-1}\left[v_k\Delta t + \tfrac{1}{2}g\Delta t^2 + \tfrac{1}{2}R_k\big(a_k - b^a_k - \eta^{ad}_k\big)\Delta t^2\right] \tag{25}$$

where $(.)_j = (.)\big(t_j\big)$ and $\Delta t_{ij} = \sum_{k=i}^{j-1}\Delta t$. Although (23) to (25) can provide the UAV's position, velocity and orientation, they are not applicable to online problems. If the linearization point at a time $t_i$ changes, all of the computations related to summations must be calculated again. To overcome this problem, we can apply the idea of the relative motion increments that is widely used in the literature (Lupton and Sukkarieh 2012):

$$\Delta R_{ij} = R_i^T R_j = \prod_{k=i}^{j-1} exp\big(\big(\omega'_k - b^g_k - \eta^{gd}_k\big)\Delta t\big) \tag{26}$$

$$\Delta v_{ij} = R_i^T\big(v_j - v_i - g\Delta t_{ij}\big) = \sum_{k=i}^{j-1}\Delta R_{ik}\big(a_k - b^a_k - \eta^{ad}_k\big)\Delta t \tag{27}$$

$$\Delta p_{ij} = R_i^T\left(p_j - p_i - v_i\Delta t_{ij} - 12\sum_{k=i}^{j-1}g\Delta t^2\right)$$
$$= \sum_{k=i}^{j-1}\left[v_k\Delta t + 12g\Delta t^2 + 12R_k\big(a_k - b^a_k - \eta^{ad}_k\big)\Delta t^2\right] \tag{28}$$

It should be highlighted that (27) and (28) can be directly calculated throughout all of the IMU observations between $k = i$ and $k = j$, so that the changes in the UAV states at the time $t_i$ do not have any adverse effect on results. Equations (26) to (28) can be utilized as a MAP estimator. However, this formulation will encounter the noise terms that increase the complexity of the computation in calculating MAP. To address this

noise mixing problem, the Rodrigues formula is utilized, and noise terms are isolated as follows (Dellaert and Kaess 2017):

$$\exp(\varphi + \delta\varphi) \approx \exp(\varphi)Exp\big(J_r(\varphi)\delta\varphi\big) \tag{29}$$

where $J_r(\emptyset)$ denotes the right Jacobian of the 3D rotation group. In other words, one can accumulate additive increments in the tangent space and multiplicative increments on the right-hand side of (29) to calculate the overall resultant. For small noises in the rotation matrix defining the vehicle's orientation, using the above formula will result in Eq. (30).

$$\Delta R_{ij} = \Delta\tilde{R}_{ij}Exp\big(-\delta\varphi_{ij}\big) \tag{30}$$

In (30), $\Delta\tilde{R}_{ij} = \prod_{k=i}^{j-1} Exp\big((\tilde{\omega}_k - b_i^g)\Delta t\big)$ and $\delta\phi_{ij}$ are representing the noisy increment term. By substituting (30) in the expression of (27) and neglecting high-order terms and repeating it for the other (26) and (28), Eqs. (31) to (33) are achieved (Forster et al. 2016):

$$\Delta\tilde{R}_{ij} = R_i^T R_j Exp\big(\delta\varphi_{ij}\big) \tag{31}$$

$$\Delta\tilde{v}_{ij} = R_i^T\big(v_j - v_i - g\Delta t_{ij}\big) + \delta v_{ij} \tag{32}$$

$$\Delta\tilde{p}_{ij} = R_i^T\Big(p_j - p_i - v\Delta t_{ij} - \tfrac{1}{2}g\Delta t_{ij}^2\Big) + \delta p_{ij} \tag{33}$$

$[\delta\phi_{ij}, \delta v_{ij}, \delta p_{ij}]$ is a random noise vector. The advantage of using (31) to (33) is that the straightforward definition of the log-likelihood is possible by defining associated noise distribution.

## Step 2: adding magnetometer measurements to graph

As a result of the magnetometer's lightweight design and relative reliability, this sensor is widely used in UAVs to measure the earth's magnetic field as an auxiliary measurement in navigation systems. This study utilizes magnetic field measurements to extract the relative position. The model of the magnetometer can be described as follows (Chesneau et al. 2017):

$$m^b = C_n^b.m^n + n_m(t) \tag{34}$$

$m^b$ is a $3 \times 1$ vector representing the magnetic field in body coordinates, $m^n$ is a $3 \times 1$ vector representing the magnetic field in navigation coordinate and $n_m$ is the Gaussian white noise of the magnetic measurements.

## Step 3: adding GNSS measurements to graph

The GNSS receiver can provide the UAV's navigation system with the position and velocity. The GNSS measurement model in the world frame is given by Eq. (35).

$$Z_{GPS}(t) = \begin{bmatrix} p^W(t) + n_{pg}(t) \\ v^W(t) + n_{vg}(t) \end{bmatrix} \tag{35}$$

$p^W(t)$ and $v^W(t)$ are denoting the position and velocity of the UAV in the world frame. Furthermore, $n_{pg}(t)$ and $n_{vg}(t)$ are the Gaussian white noises of the position and velocity measurements, respectively.

## Step 4: adding barometer measurements to graph

The measurements of the barometric altitude can be obtained by measuring barometric pressure and converting it to altitude:

$$Z_{Baro}(t) = -d(t) + n_{pres}(t) \tag{36}$$

In (36), $d(t)$ and $n_{pres}(t)$ represent the measured barometric altitude and the Gaussian white noise of the air sensor, respectively. It is worth noting that the measured altitudes by the barometric sensor and GNSS are related to the sea level and the World Geodetic System 1984 (WG84) coordinate frame, respectively. Consequently, the altitude must be changed into an actual altitude before data fusion.

## Step 5: forming sliding window on graph

Since the sampling frequency of the GNSS is much lower than other sensors, the width of the sliding window used for marginalization can be set equal to the time passed between two GNSS consecutive measurements (Fig. 8). Further, the other sensors must be synchronized with the GNSS. Another main point that needs to be highlighted is that the measurements of the sensors calculated in the body frame must be transformed to the world frame before adding them to the optimization graph to form and solve optimization in a single coordinate, namely the word frame.

## Step 6: solving the optimization problem by factor graph

To solve the nonlinear optimization problem, repeated linearization within a standard Newton–Gaussian optimizer is utilized. According to (5), the optimization problem can be modeled as (37) (Indelman et al. 2013).

$$\overline{\delta x} = \underset{\Delta X}{\arg\min}\, r(\hat{X}) - J(\hat{X})\Delta X \tag{37}$$

$X$ denotes the set of all variables and $\hat{X}$ is an initial estimate of $X$. $J(\hat{X})$ and $r(\hat{X})$ demonstrate the sparse Jacobian matrix at the current linearization point and the residual given all measurements, respectively. Once the updated value of the $\Delta X$ is calculated, the value of $X + \Delta X$ is obtained and considered as an initial estimate for the next round of iteration. The cost function of $\Delta X$ is defined as,

**Fig. 8** Sliding window used in the graph-based proposed method. Multiple sensors with different frequencies can be utilized in navigation due to the employed factor graph method. A sliding window is added to the algorithm to marginalize prior measurements to attain a more time-efficient method, thus solving a less complicated optimization problem for every added measurement
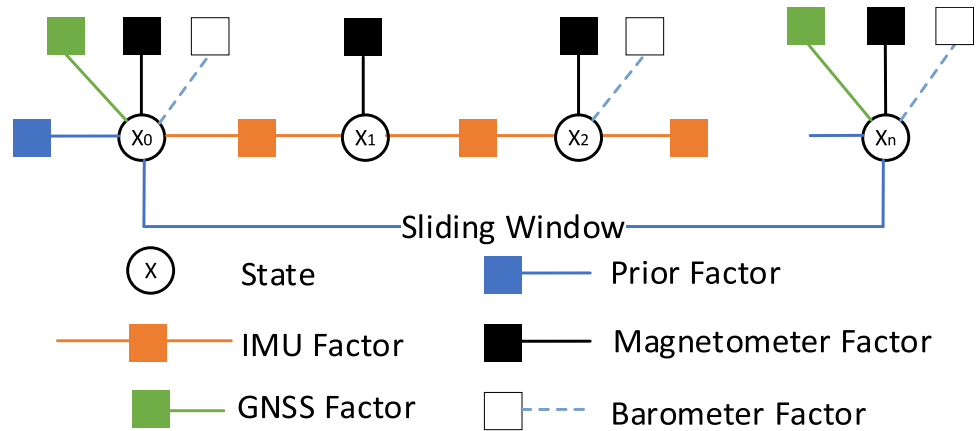


$$\min_{X} \left\{ (H_p \delta X_r - b_p) + \sum_{k \in B} r_B \left( \hat{Z}^{b_k}_{b_{k+1}}, X \right)_{P_B} + \sum_{k \in G} r_G \left( \hat{Z}^{g_k}_{g_{k+1}}, X \right)_{PG} \right.$$
$$\left. + \sum_{k \in M} r_M \left( \hat{Z}^{m_k}_{m_{k+1}}, X \right)_{P_M} + \sum_{k \in A} r_A \left( \hat{Z}^{a_k}_{a_{k+1}}, X \right)_{P_A} \right\} \tag{38}$$

where $H_P$ and $b_P$ are related to the prior information that the marginalization operation can compute. The symbols $r_B$, $r_G$, $r_M$ and $r_A$ illustrate the errors caused by the IMU, GNSS, magnetometer and air sensor measurements, respectively.

### Step 7: Marginalization

As a result of a large number of states, the marginalization method is utilized to avoid high computational complexity while preserving primary information. The marginalization method uses the previous information as a prior term so that the sets of the marginalized and reminder states are specified as $X_M$ and $X_R$, respectively. According to (38), (39) can be derived by applying the Newton–Gauss formula,

$$J(x)^T P^{-1} J(x) \delta x = -J(x)^T P^{-1} r(x) \tag{39a}$$

$$H = J(x)^T P^{-1} J(x) \tag{39b}$$

$$b = -J(x)^T P^{-1} r(x) \tag{39c}$$

Equation (40) can be obtained by sorting marginalized and remaining states into groups.

$$\begin{bmatrix} H_{MM} & H_{MR} \\ H_{RM} & H_{RR} \end{bmatrix} \begin{bmatrix} \delta X_M \\ \delta X_R \end{bmatrix} = \begin{bmatrix} b_M \\ b_R \end{bmatrix} \tag{40}$$

The marginalization can be performed by the Schur complement (Sibley et al. 2010) as follows:

$$\begin{bmatrix} I & 0 \\ -H_{RM}H_{MM}^{-1} & I \end{bmatrix} \begin{bmatrix} H_{MM} & H_{MR} \\ H_{RM} & H_{RR} \end{bmatrix} \begin{bmatrix} \delta X_M \\ \delta X_R \end{bmatrix} = \begin{bmatrix} I & 0 \\ -H_{RM}H_{MM}^{-1} & I \end{bmatrix} \begin{bmatrix} b_M \\ b_R \end{bmatrix} \rightarrow$$
$$\begin{bmatrix} H_{MM} & H_{MR} \\ 0 & -H_{RM}H_{MM}^{-1}H_{MR} + H_{RR} \end{bmatrix} \begin{bmatrix} \delta X_M \\ \delta X_R \end{bmatrix} = \begin{bmatrix} b_M \\ -H_{RM}H_{MM}^{-1}b_M + b_r \end{bmatrix} \rightarrow$$
$$(-H_{RM}H_{MM}^{-1}H_{MR} + H_{RR})\delta X_R = -H_{RM}H_{MM}^{-1}b_M + b_r \tag{41a}$$

$$H_P = -H_{RM}H_{MM}^{-1}H_{MR} + H_{RR} \tag{41b}$$

$$b_P = -H_{RM}H_{MM}^{-1}b_M + b_r \tag{41c}$$

The above equation implies that the new prior $H_p$ and $b_p$ are achievable such that the information of the marginalized states is preserved and can be converted into prior terms; therefore, the problem of the MAP (5) can be rewritten as follows:

$$\hat{X}_i^{MAP} = \arg \min_{X_i} \left\{ \sum_{t=m}^{n} \sum_{k \in S} \left\{ Z_t^k - h_t^k (X_i)^2_{\Omega_t^k} + (H_p \delta X_r - b_p) \right\} \right\} \tag{42}$$

$S$ shows the set of the available measurements. Finally, in the last step of the proposed method, the algorithm will loop back to step two if a new measurement is received.

## Experiment description and results

This section compares the Factor Graph Optimization (FGO) algorithm with the conventional EKF method widely used in the literature. Many aspects need to be considered for this evaluation. The states used to verify the proposed method consist of the position (latitude, longitude, altitude), and velocity components in the north, east, down (NED) Coordinate system $(v_N, v_E, v_d)$. The vertical speed and gamma angle are essential information in the landing phase; they are calculated and extracted through the position and velocity direction. All calculations are carried out in the body system.

**Table 1** Sensor specification of the SPATIAL INS. This table presents the specifications of the sensors used in SPATIAL INS

| Parameters | Gyroscope | Accelerometer | Magnet | Barometer |
|---|---|---|---|---|
| Dynamic range | $\pm 2000$ deg/s$pm 2000$ deg/s | $\pm 16$ g | $\pm 4$ G $\pm 8$ G | 10–120 kPa |
| Nonlinearity | <0.05% | <0.05% | <0.08% | – |
| Scale factor stability | <0.05% | <0.06% | <0.09% | – |
| Noise density | $0.004^0 \frac{\sqrt{\text{Hz}}}{s}$ | $100 \frac{\text{ug}}{\sqrt{\text{Hz}}}$ | $210 \frac{\text{ug}}{\sqrt{\text{Hz}}}$ | $0.56 \frac{\text{Pa}}{\sqrt{\text{Hz}}}$ |
| Bandwidth | 400Hz | 400Hz | 100Hz | 50 Hz |

As it can be seen, there are multiple sensors employed in this INS to aid the navigation with additional measurements to attain accurate navigation
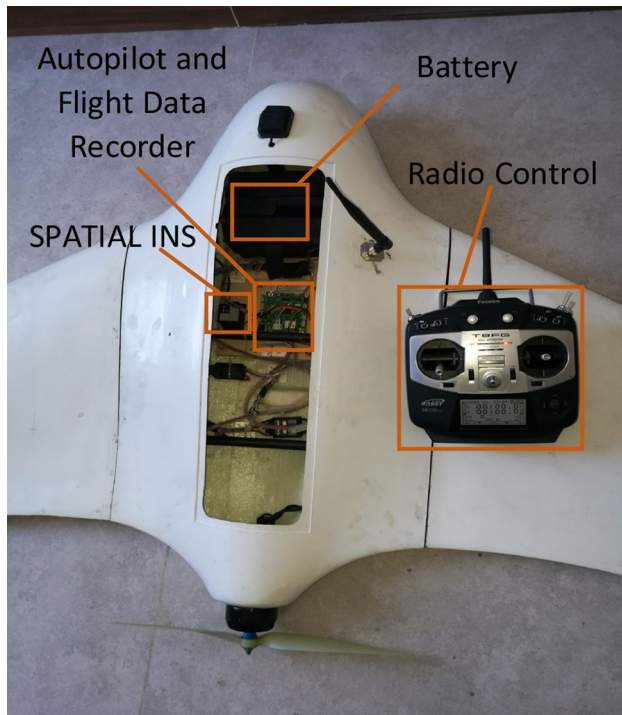


**Fig. 9** Picture of X8Skywalker UAV used to collect the dataset. The UAV is equipped with the SPATIAL INS, autopilot, and a flight data recorder to carry out the field tests

**Table 2** GNSS specification of the SPATIAL INS

| Sensor | Parameter | Value |
|---|---|---|
| SPATIAL | Horizontal position accuracy | 2.0 m |
| | Vertical position accuracy | 3.0 m |
| | Velocity accuracy | 0.05 m/s |
| | Roll and pitch accuracy (dynamic) | 0.2 deg |
| | Output data rate | 50–500 Hz |
| | Heading accuracy (dynamic, magnetic only) | 0.8 deg |
| GNSS | Supported navigation systems | GPS L1 GLONASS L1 GALILEO E1 |
| | Update rate | 10 Hz |
| | Horizontal position accuracy | 2.5 m |
| | Velocity accuracy | 0.05 m/s |
| | Heading accuracy (Dynamic with GNSS) | 0.2 deg |

Used INS supports multiple navigation systems like GPS L1, GLONASS L1, and GALILEO E1

Coordinates are converted to navigation coordinates north, east, down (NED) for plotting.

To compare the FGO and EKF methods, the SPATIAL INS, a high-efficacy integrated navigation system belonging to the Advanced Navigation Company (https://www.advancednavigation.com/solutions/spatial), is chosen as a reference measurement. Also, all sensors characteristics of the SPATIAL INS are shown in Table 1. We collected the flight dataset by utilizing a SkywalkerX8 UAV. Figure 9 depicts the platform of the X8 with the utilized sensors and equipment. The flight time is about seven minutes, and the X8 experienced all sorts of maneuvers during the flight test, including climb, descent, turn and cruise. The navigation specification of SPATIAL and the GNSS characteristics are shown in Table 2. Also, as depicted in Fig. 10, UAV's navigation is assessed through starting position ($P_s$) to the final position ($P_f$). We selected two separate two-minute regions to examine the proposed algorithm in more detail, between 20 and 140 s in the first region (REG1) and between 200 and 320 s in the second region (REG2). To demonstrate the superiority of the proposed algorithm, we compared position and velocity estimates from FGO and EKF with the SPATIAL INS (reference) by computing various statistical measures and execution time. The results given in this section have been achieved on a windows eight system using Intel Core i5, 2.5 GHz, 12G Random Access Memory (RAM), and MATLAB R2019b.

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left( t_{Ri} - t_{Ei} \right)^2} \tag{43}$$

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^{N} |t_{Ri} - t_{Ei}| \tag{44}$$

**Fig. 10** Trajectory of the X8 flight. The flight starts at position $P_s$ and finished at $P_f$. Two regions of interests Region 1 and Region 2 are selected to evaluate our proposed method and compare it to the EKF algorithm
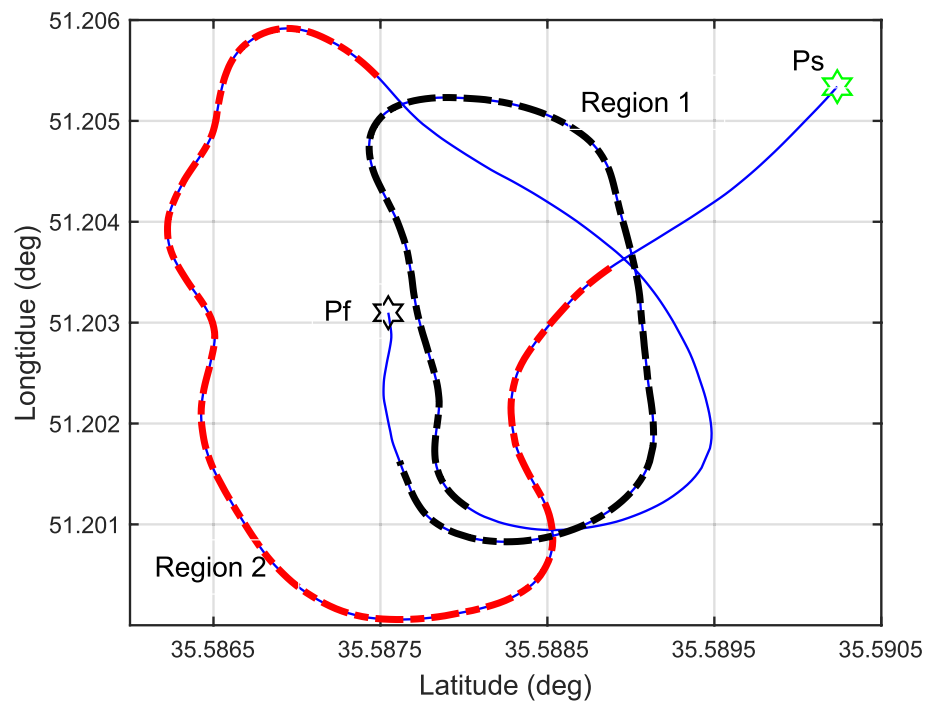


$$\text{Std} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left( \mu - \left( t_{Ri} - t_{Ei} \right) \right)^2} \tag{45}$$

In above equation, $t_{Ri}$, $t_{Ei}$ and $\mu$ denote the reference, estimated values, and average error, respectively. Also, as shown in Tables 3, 4, 5, the accuracy of the reference values is reasonably good so that they cannot have any adverse effect on the final results. Figures 11, 12, 13, 14 illustrate the more excellent performance of the FGO algorithm in estimating position and velocity during the mentioned regions. The red, blue and green lines show the reference, FGO (proposed algorithm) and EKF, respectively.

As shown in Fig. 11, although the trajectory of the X8 fight consists of rich maneuvers with many changes, the proposed FGO system can estimate flight trajectory more

**Table 3** RMSE results of the proposed FGO and traditional EKF frameworks in the flight test

| Navigation state | RMSE | | | |
|---|---|---|---|---|
| | REG1 | | REG2 | |
| | EKF | FGO | EKF | FGO |
| Latitude (deg) | 4.5750e-5 | 2.2852e-5 | 3.8553e-5 | 2.1243e-5 |
| Longitude (deg) | 6.1929e-5 | 3.4472e-5 | 6.2713e-5 | 4.9481e-5 |
| Height (m) | 4.6302 | 1.4892 | 6.6899 | 2.7286 |
| V-north (m/s) | 4.3652 | 1.9485 | 3.2251 | 1.8005 |
| V-east (m/s) | 6.1776 | 4.3366 | 5.1255 | 4.9386 |
| V-down (m/s) | 7.7622 | 0.8148 | 7.5703 | 0.5093 |

**Table 4** MAE results of the proposed FGO and traditional EKF frameworks in the flight test

| Navigation state | MAE | | | |
|---|---|---|---|---|
| | REG1 | | REG2 | |
| | EKF | FGO | EKF | FGO |
| Latitude (deg) | 0.0060 | 0.0045 | 0.0055 | 0.0042 |
| Longitude (deg) | 0.0072 | 0.0054 | 0.0073 | 0.0067 |
| Height (m) | 1.9482 | 1.1431 | 2.5108 | 1.6118 |
| V-north (m/s) | 1.8592 | 1.2944 | 1.6027 | 1.2277 |
| V-east (m/s) | 2.1608 | 1.9536 | 2.0490 | 2.1506 |
| V-down (m/s) | 2.5014 | 0.8000 | 2.4561 | 0.6223 |

**Table 5** Std results of the proposed FGO and traditional EKF frameworks in the flight test

| Navigation state | Std. | | | |
|---|---|---|---|---|
| | REG1 | | REG2 | |
| | EKF | FGO | EKF | FGO |
| Latitude (deg) | 0.0019 | 6.3310e-4 | 0.0021 | 0.0010 |
| Longitude (deg) | 0.0019 | 6.2026e-4 | 0.0021 | 0.0010 |
| Height (m) | 4.6301 | 1.4889 | 6.6884 | 2.7280 |
| V-north (m/s) | 4.3639 | 1.9487 | 3.2250 | 1.8003 |
| V-east (m/s) | 6.1763 | 4.3361 | 5.1255 | 4.9379 |
| V-down (m/s) | 7.7611 | 0.8147 | 7.5688 | 0.5088 |

**Fig. 11** Longitude and latitude comparison for the first region (origin latitude = 35.590 and longitude = 51.2055 degrees). Estimated latitude and longitude of the FGO (blue) and EKF (green) compared to the reference (red) trajectory in the first region (**a**). The FGO and EKF longitudinal (**b**) and latitudinal (**c**) errors with respect to the reference. The error figures show that the FGO error is smaller than those of the EKF

**Fig. 12** Velocity and height estimations for the first region. North, east, down velocities, and height of the FGO (blue) and EKF (green) compared to the reference trajectory (red) in the first region. The computed east and north velocity components show the higher accuracy of the suggested algorithm during prolonged intervals. It should be noted that distinct gravitational models and mismatched altitude data of GNSS and barometer can cause fluctuations in the vertical velocity $(v_d)$ estimations

**Fig. 13** Longitude and latitude comparison for the second region (origin latitude = 35.590 and longitude = 51.2055 degrees). Results of the latitude and longitude of the FGO (blue) and EKF (green) compared to the reference (red) trajectory in the second region (**a**). The FGO and EKF longitudinal (**b**) and latitudinal (**c**) errors with respect to the reference. Just like Fig. 11, the FGO algorithm exhibits better performance

**Fig. 14** Velocity and height estimations for the second region. North, east, down velocities, and height of the FGO (blue) and EKF (green) solutions compared to the reference trajectory (red) in the second region
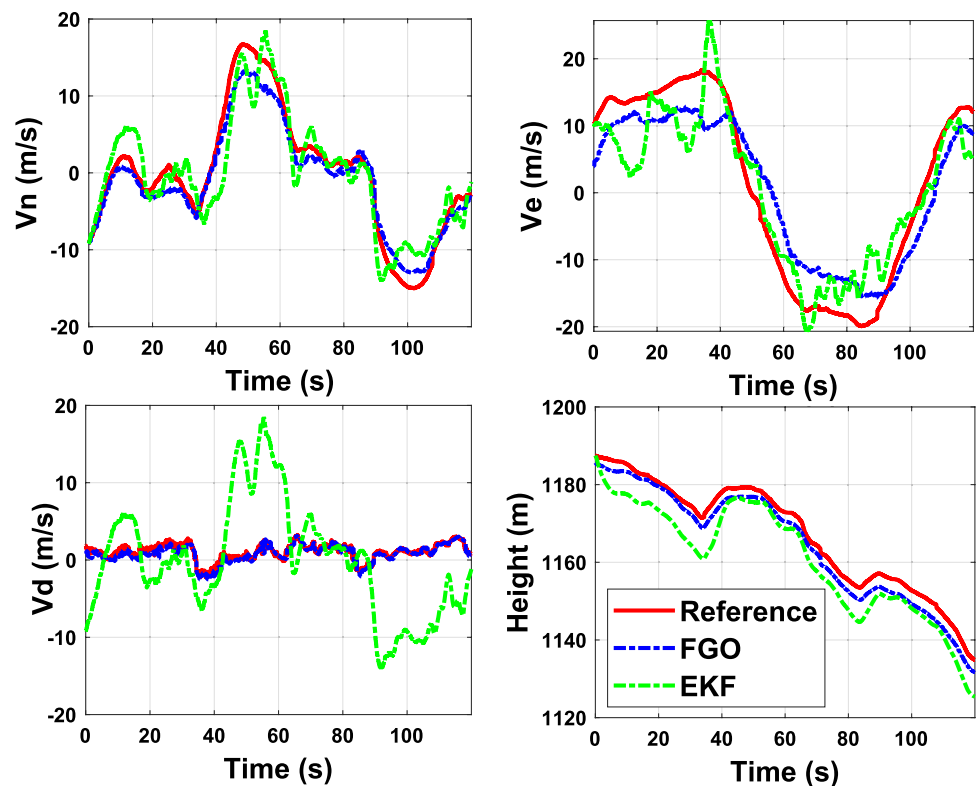


accurately than the conventional EKF. To verify the superiority of the proposed FGO system, we give the RMSE, MAE and Std. results of our method and conventional EKF in Tables 3, 4 and 5. As these tables illustrate, the proposed FGO system has a superior performance in all navigation states, especially in terms of $V_n$, $V_d$, and height compared to the EKF. The navigation accuracy of the UAV is the most crucial factor for having a successful landing phase. So, our system can fulfill this requirement successfully. The most important feature of the window-based optimization methods is smoothing compared to the EKF, clearly seen in Figs. 11, 12 13, 14.

We also investigated the execution time of our method to evaluate the feasibility of implementing the suggested FGO system. For this purpose, we considered three cases with various sensors, including (1) IMU and GNSS, (2) IMU, GNSS and magnetometer, and (3) IMU, GNSS, magnetometer, and barometer operation modes. Furthermore, the number of samples of the IMU, GNSS, barometer and magnetometer sensors utilized to calculate the overall execution time is 18000, 1800, 9000 and 18,000, respectively. The resultant execution times of the proposed FGO

system are given in Table 6. Maximum execution time is related to the optimization operation, which is responsible for 75% of the runtime. Finally, we should mention that the execution time can be improved by utilizing high-speed computers with higher frequency cores and larger RAM.

## Conclusions

This study proposed a navigation algorithm to improve the UAV's navigation accuracy in the landing phase. We introduced a tightly-coupled algorithm based on factor graph optimization. For this purpose, first, the integration method of the IMU observations in the body frame between two consecutive GNSS measurements was developed. In addition to the GNSS receiver, we utilized magnetometer and barometer sensors to add more measurements to the graph. The most important feature of the factor graph approach in navigation problems is its convenience in adding or removing various sensors with different sampling frequencies. However, finding optimal navigation solutions using factor graphs is computationally intensive. By this means, we implemented a sliding window and

**Table 6** Execution time of the proposed FGO system for three different modes utilizing various sensors

| Sensors | IMU + GNSS | IMU + GNSS + magnet | IMU + GNSS + magnet + barometer |
|---|---|---|---|
| Time (s) | 6.8 | 12.4 | 15.6 |

marginalized prior measurements so the optimization could be applicable for real-time navigation. Furthermore, to verify the performance of the suggested FGO system, we utilized a dataset collected by X8skywalker UAV. The field test of the FGO system was compared to the reference and EKF systems by utilizing and commercially available high-performance INS system as a reference measurement. The results show that the proposed system outperforms the traditional EKF navigation, especially in position and velocity values.

# References

Carlone L, Kira Z, Beall C, Indelman V, Dellaert F (2014) Eliminating conditionally independent sets in factor graphs: a unifying perspective based on smart factors. In: 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, June 1–7, pp. 4290–4297

Chesneau CI, Hillion M, Hullo JF, Thibault G, Prieur C (2017) Improving magneto-inertial attitude and position estimation by means of a magnetic heading observer. In: 2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Sapporo, Japan, September 18–21, pp. 1–8

Dai B, He Y, Yang L, Su Y, Yue Y, Xu W (2020) SIMSF: a scale insensitive multi-sensor fusion framework for unmanned aerial vehicles based on graph optimization. IEEE Access 8:118273–118284

Dellaert F, Kaess M (2017) Factor graphs for robot perception. Found Trends Robot 6(1–2):1–139

Eltahier MMA, Hamid K (2017) Review of instrument landing system. IOSR J Electr Commun Eng 12(1):106–113

Forster C, Carlone L, Dellaert F, Scaramuzza D (2016) On-manifold pre-integration for real-time visual-inertial odometry. IEEE Trans Rob 33(1):1–21

Groves PD (2015) Principles of GNSS, inertial, and multi sensor integration navigation systems. IEEE Aerosp Electron Syst Mag 30(2):26–27

Gu X, Zhang F, Xu J, Yuan Q, Ma H, Liu X, Li Z (2019) Graph optimization based long-distance GPS/IMU integrated navigation. In: IEEE 2019 Chinese Control Conference (CCC), pp. 3976–3981

Indelman V, Williams S, Kaess M, Dellaert F (2013) Information fusion in navigation systems via factor graph based incremental smoothing. Robot Auton Syst 61(8):721–738

Jizhou LAI, Shiyu BAI, Xiaowei XU, Pin L (2019) A generic plug-and-play navigation fusion strategy for land vehicles in GNSS-denied environment. Trans Nanjing Univ Aeronaut Astronaut 36(2):197–204

Jones ES, Soatto S (2011) Visual-inertial navigation, mapping and localization: a scalable real-time causal approach. Int J Robot Res 30(4):407–430

Kaess M, Johannsson H, Roberts R, Ila V, Leonard JJ, Dellaert F (2012) ISAM2: incremental smoothing and mapping using the Bayes tree. Int J Robot Res 31(2):216–235

Kschischang FR, Frey BJ, Loeliger HA (2001) Factor graphs and the sum-product algorithm. IEEE Trans Inf Theory 47(2):498–519

Li W, Cui X, Lu M (2018) A robust graph optimization realization of tightly coupled GNSS/INS integrated navigation system for urban vehicles. Tsinghua Sci Technol 23(6):724–732

Lupton T, Sukkarieh S (2012) Visual-inertial-aided navigation for high-dynamic motion in built environments without initial conditions. IEEE Trans Rob 28(1):61–76

Lynen S, Achtelik MW, Weiss S, Chli M, Siegwart R (2013) A robust and modular multi-sensor fusion approach applied to MAV navigation. In: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, November 1–7, pp. 3923–3929

Mascaro R, Teixeira L, Hinzmann T, Siegwart R, Chli M (2018) GOMSF: Graph-Optimization based Multi-Sensor Fusion for robust UAV pose estimation. In: 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, May 21–25, pp. 1421–1428.

Mourikis AI, Roumeliotis SI (2007) A multi-state constraint Kalman filter for vision-aided inertial navigation. In: Proceedings 2007 IEEE International Conference on Robotics and Automation, Roma, Italy, April 10–14, pp. 3565–3572

Mu X, Chen J, Zhou Z, Leng Z, Fan L (2018) Accurate initial state estimation in a monocular visual–inertial SLAM system. Sensors 18(2):506

Mur-Artal R, Tardós JD (2017) Visual-inertial monocular SLAM with map reuse. IEEE Robot Autom Lett 2(2):796–803

Nezhadshahbodaghi M, Mosavi MR (2021) A loosely-coupled EMD-denoised stereo VO/INS/GPS integration system in GNSS-denied environments. Measurement 183:109895

Nezhadshahbodaghi M, Mosavi MR, Hajialinajar MT (2021) Fusing denoised stereo visual odometry, INS and GPS measurements for autonomous navigation in a tightly coupled approach. GPS Solut 25(2):1–18

Nourmohammadi H, Keighobadi J (2017) Decentralized INS/GNSS system with MEMS-grade inertial sensors using QR-factorized CKF. IEEE Sens J 17(11):3278–3287

Qin T, Cao S, Pan J, Shen S (2019) A general optimization-based framework for global pose estimation with multiple sensors. arXiv preprint, Cornel University, January 11, https://arxiv.org/abs/1901.03642.

Rogers RM (2007) Applied mathematics in integrated navigation systems. American Institute of Aeronautics and Astronautics, USA

Shen S, Mulgaonkar Y, Michael N, Kumar V (2014) Multi-sensor fusion for robust autonomous flight in indoor and outdoor environments with a rotorcraft MAV. In: 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, June 1–7, pp. 4974–4981

Sibley G, Matthies L, Sukhatme G (2010) Sliding window filter with application to planetary landing. J Field Robot 27(5):587–608

Titterton D, Weston JL (2004) Strap down inertial navigation technology, 2nd edn. Institution of Engineering Technology, Stevenage

Wei X, Li J, Zhang D, Feng K (2021) An improved integrated navigation method with enhanced robustness based on factor graph. Mech Syst Signal Process 155:107565

Wen W, Pfeifer T, Bai X, Hsu LT (2020) It is time for factor graph optimization for GNSS/INS integration: comparison between FGO and EKF. arXiv preprint, https://arxiv.org/abs/2004.10572.

Xu J, Yang G, Sun Y, Picek S (2021) A multi-sensor information fusion method based on factor graph for integrated navigation system. IEEE Access 9:12044–12054

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Sina Taghizadeh** received his B.S. degree in Physics from Qom University in 2003 and M.S. degree in Electronic Engineering from MUT University in 2006, Tehran, Iran. He is currently a Ph.D. student in the Computer Engineering Department, Amirkabir University of Technology, Tehran, Iran. His research interests include artificial intelligence, INS, visual-inertial navigation, and deep learning.

**Reza Safabakhsh** received his electrical engineering B.S. from the Sharif University of Technology, Tehran, Iran (1976) and the M.S. (1980) and Ph.D. degrees (1986) in electrical and computer engineering from the University of Tennessee, Knoxville. He is currently a professor, the director of the Deep Learning and Computer Vision Laboratories, and a member of the IEEE and several honor societies, including ΦΚΦ and HKN.

**Mohsen Nezhadshahbodaghi** received his B.S. and M.S. degrees in Electronic Engineering from Shahid Bahonar University of Kerman in 2016 and Iran University of Science and Technology (IUST) in 2018, Tehran, Iran. He is currently a Ph.D. student at the IUST Department of Electrical Engineering. His research interests include signal processing, artificial intelligence, INS, visual odometry, and GPS applications.

**Mohammad Reza Mosavi** received his B.S., M.S., and Ph.D. degrees in Electronic Engineering from Iran University of Science and Technology (IUST), Tehran, Iran, in 1997, 1998, and 2004, respectively. He is currently a faculty member (full professor) of the Department of Electrical Engineering of IUST. He is the author of more than 450 scientific publications in journals and international conferences in addition to 12 academic books. His research interests include circuits and systems design.