

# Field validation of GNSS-independent positioning enhancement using a wearable ultra-stable quantum magnetometer supplemental

Stirling Scholes<sup>1\*</sup>, Dominic Hunter<sup>1</sup>, Courtney Dyer<sup>1</sup>,  
Marcin Mrozowski<sup>1</sup>, Allan McWilliam<sup>1</sup>, Phoebe Utting<sup>1</sup>,  
David Burt<sup>2</sup>, Paul F. Griffin<sup>1,3</sup>, James P. McGilligan<sup>1,3</sup>,  
Erling Riis<sup>1,3</sup>, Stuart J. Ingleby<sup>1,3</sup>

<sup>1</sup>Department of Physics, SUPA, University of Strathclyde, Glasgow,  
G4 0NG, United Kingdom.

<sup>2</sup> Kelvin Nanotechnology, University of Glasgow, Glasgow, G12 8LS,  
United Kingdom.

<sup>3</sup> Quantrologee, Glasgow, G12 8QQ, United Kingdom.

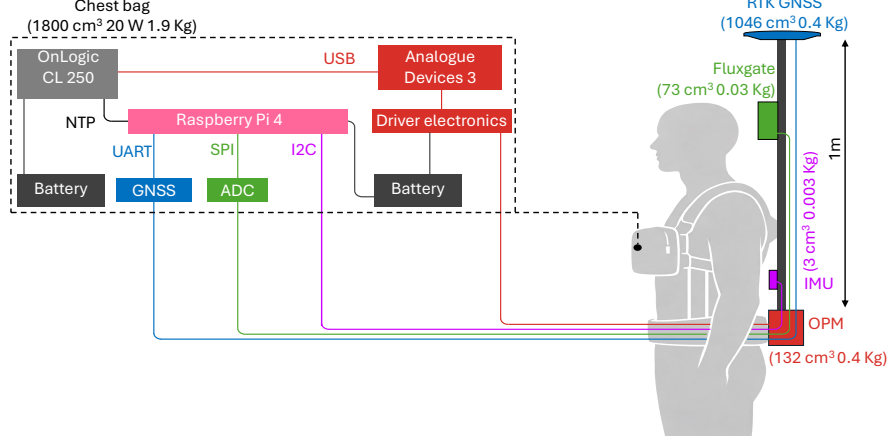
\*Corresponding author(s). E-mail(s): [s.scholes@strath.ac.uk](mailto:s.scholes@strath.ac.uk);

Contributing authors: [d.hunter@strath.ac.uk](mailto:d.hunter@strath.ac.uk);  
[courtney.dyer@strath.ac.uk](mailto:courtney.dyer@strath.ac.uk); [marcin.mrozowski@strath.ac.uk](mailto:marcin.mrozowski@strath.ac.uk);  
[paul.griffin@strath.ac.uk](mailto:paul.griffin@strath.ac.uk); [james.mcgilligan@strath.ac.uk](mailto:james.mcgilligan@strath.ac.uk);  
[e.riis@strath.ac.uk](mailto:e.riis@strath.ac.uk); [stuart.ingleby@strath.ac.uk](mailto:stuart.ingleby@strath.ac.uk);

**Keywords:** Positioning, Navigation, Quantum sensing, Optically pumped magnetometers

## 1 Wearable system configuration and base station

Figure 1 shows the block diagram for the wearable sensor system. Data logging was achieved using a Raspberry Pi 4 and an OnLogic CL250. The computers were connected over ethernet and the Network Timing Protocol (NTP) was used to maintain time synchronicity with the Pi acting as an NTP server. The GNSS signal was processed by the ArduSimple board connected to the Pi using UART. The Pulse-Per-Second (PPS) timing signal from the GNSS board was passed to the Pi



**Fig. 1:** Companion to Fig. 1 in the main manuscript showing the block diagram for the electronic components of the wearable system.

and used to discipline the Pi’s internal clock. The analogue signal from the fluxgate was digitised by a Waveshare High-Precision Pi-Hat Analogue to Digital Converter (ADC) connected over the SPI-interface. The IMU was connected directly to the Pi via the I2C protocol. The OPM was connected to its controlling electronics and communicated with the CL250 via an AD3 and USB. By separating the computing load, the diversity of interfaces required to communicate with all devices could be achieved while maximising the measurement bandwidth for the OPM. Two battery packs were used due to the maximum current delivery limitations of the packs.

Figure 2 shows the base station used during the trial. The tripod was placed outside of the survey area to prevent magnetic interference. The components of the base station are a repeat of the Raspberry Pi, GNSS, and fluxgate combination shown in Fig. 1. The base station is powered by a Jackery Box portable power unit.

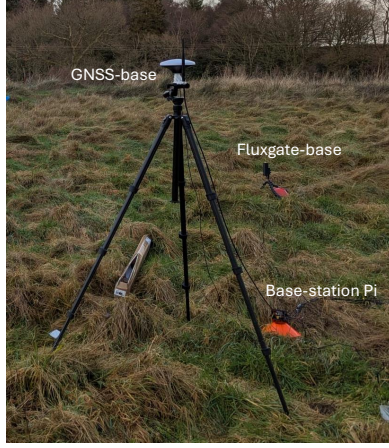
## 2 Sensor calibration procedure

Using the fluxgate, the hard-iron and soft-iron error from the platform is characterised by an ellipsoid of the form [1, 2]:

$$Ax^2 + By^2 + Cz^2 + 2Dxy + 2Exz + 2Fyz + 2Gx + 2Hy + 2Iz + J = 0, \quad (1)$$

where the  $x, y, z$  coordinates correspond to North, East, Down. Expressing the coefficients as a vector  $\alpha$ , a system of equations  $u$  is given by,

$$(\alpha^T * \alpha) u = \alpha^T N. \quad (2)$$



**Fig. 2:** A photograph of the base station in the field.

$N$  is the magnitude of the field measured by the fluxgate. Once solved, the coefficients from  $u$  can be written as a matrix  $\beta$ ,

$$\beta = \begin{bmatrix} u_1 + u_2 - 1 & u_3 & u_4 & u_6 \\ u_3 & u_1 - 2u_2 - 1 & u_5 & u_7 \\ u_4 & u_5 & u_2 - 2u_1 - 1 & u_8 \\ u_6 & u_7 & u_8 & u_9 \end{bmatrix} \quad (3)$$

The hard-iron error is then given by the centre offset vector  $\gamma$  of the ellipsoid,

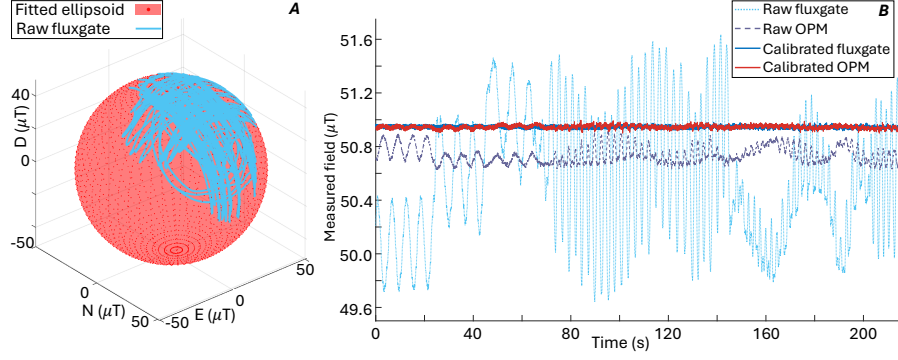
$$-\beta(1 : 3, 1 : 3) \gamma = [u_7, u_8, u_9]. \quad (4)$$

The soft-iron error  $\epsilon$  is given by,

$$\epsilon = k * [\lambda * \text{diag}(1/R) * \lambda^T]. \quad (5)$$

$k$  is the reference field magnitude,  $\text{diag}(1/R)$  is a  $3 \times 3$  diagonal matrix containing the radii  $R$ .

$$R = \sqrt{1/\text{diag}(|e|)}, \quad (6)$$



**Fig. 3:** *A* The ellipsoid fit used to correct for hard-iron and soft-iron error. The pale blue points show the uncorrected fluxgate data in an NED coordinate system. The red orb shows the fitted ellipsoid. *B* The contrast between raw and calibrated fluxgate and OPM data. The raw fluxgate and raw OPM calibration data are shown as dashed pale blue and dashed pale maroon lines respectively. The corrected fluxgate and corrected OPM data are shown as the largely overlapping blue and red lines.

where  $e$  is the eigenvalue corresponding to the eigenvector  $\lambda$  of the centre corrected matrix  $\beta$ , i.e.,

$$\begin{aligned}
 P &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \gamma_1 & \gamma_2 & \gamma_3 & 1 \end{bmatrix} \\
 Q &= P * \beta * P^T \\
 S &= Q[1:3, 1:3] / -Q_{44} \\
 Se &= \lambda e.
 \end{aligned} \tag{7}$$

The corrected fluxgate  $B_{N,E,D}$  is given by,

$$B_{N,E,D} = \epsilon * (B_{N,E,D}^{\text{raw}} - \gamma). \tag{8}$$

Figure 3 shows the calibration process for the magnetic sensors on the platform. Figure 3A shows the raw measured magnetic field vector from the uncorrected fluxgate as pale blue dots in an NED coordinate space. The corresponding raw fluxgate magnitude is shown as a function of time in Fig. 3B. In Fig. 3B both the fluxgate magnitude and raw OPM data (dashed maroon line) exhibit oscillations as a function of platform orientation, characteristic of hard-iron error. The fluctuations in the fluxgate measurement would degrade the performance of the heading-based estimator used in the particle-filter-based map matching. Further, the oscillations in the OPM data are of meaningful magnitude compared to the magnitude of the local anomalies

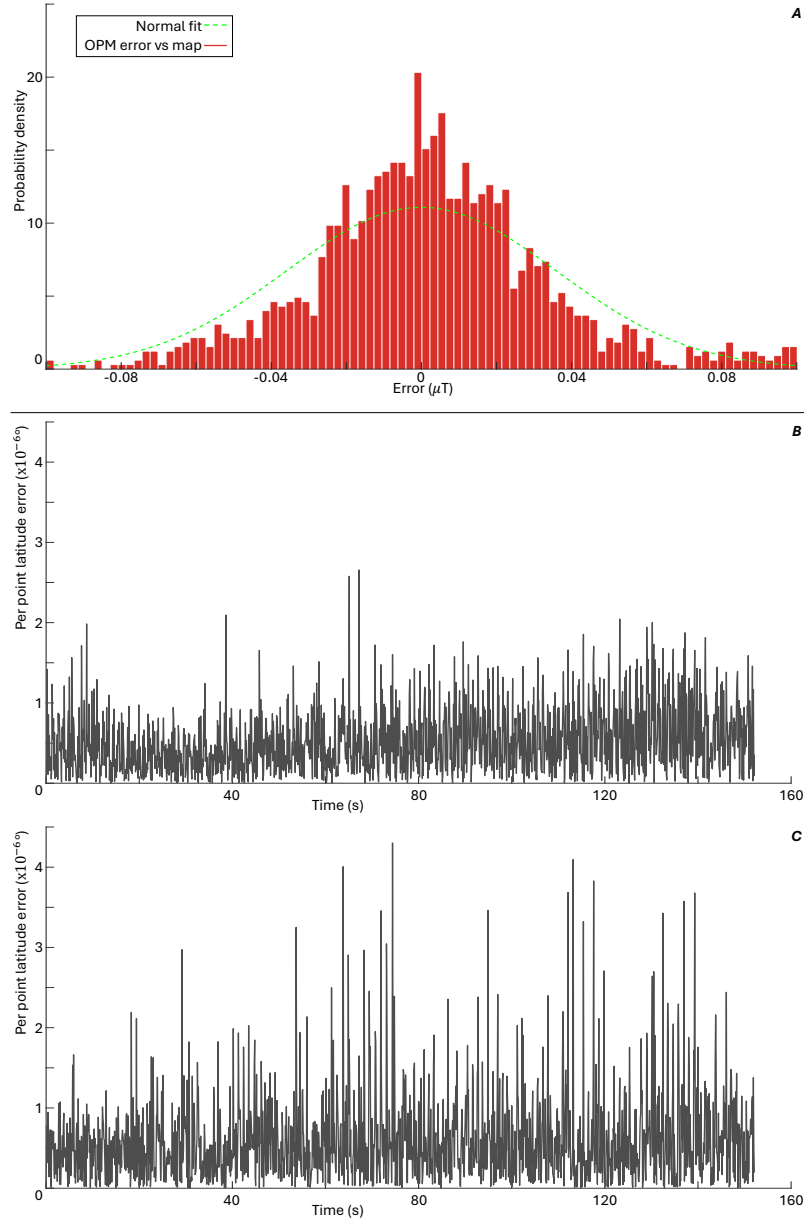
in the magnetic map (Fig. 3 of the main manuscript). Together, these factors highlight the need for effective hard and soft-iron error mitigation.

The red ellipsoid in Fig. 3A shows the 3D fit to the raw fluxgate data. The offset of the centre of the fitted ellipsoid corresponds to the hard-iron error experienced by the fluxgate while the ellipticity is related to the soft-iron error. Using the parameters of the fit, the hard- and soft-iron errors in the fluxgate can be mitigated, as shown by the calibrated fluxgate trace in Fig. 3B which is stable throughout orientation changes. Once calibrated, the fluxgate provides an attitude reference for the system. This is used to calculate a Tolles-Lawson correction for the OPM. The corrected OPM data is shown as the orange-red trace in Fig. 3B.

### 3 Algorithm calibration

Figure 4 shows the plots used to derive the measurement noise and process noise parameters for the particle filter. Figure 4A shows a histogram of the error between the measured OPM value and the map-interpolated value for the two algorithm calibration lines. The dashed green line shows a Normal fit to the data with a standard deviation corresponding to the measurement noise.

Figures 4B and C show the absolute point-to-point latitude and longitude error respectively. This was computed by first converting the GNSS ground truth coordinates recorded in the calibration lines into lists of latitude and longitude changes from each point to the next. Then, taking each GNSS point as a starting coordinate and computing the latitude and longitude change given by the heading and fixed step size estimator for a single step. The process noise is calculated as the average between the largest error observed in latitude and the largest error in longitude, ensuring the particle swarm retains sufficient diversity to absorb errors introduced by the dead reckoning estimator.



**Fig. 4:** *A* The distribution of error between the measured OPM values and interpolated map values in the algorithm calibration lines. The dashed green line shows the Normal fit associated with the measurement noise value. *B* and *C* The absolute value of the point-to-point latitude and longitude error in the calibration lines used to derive the process noise term.

## 4 Map matching algorithm

Algorithm 1 details the implementation of the particle filter used for map matching [3, 4].

---

### Algorithm 1 Magnetic map matching particle filter implementation

---

**Require:** Initial position estimate  $P_0 = [\text{Lat}_1, \text{Lon}_1]^T$ , initial process noise  $\sigma_1^0$ , number of particles  $N$ , process-noise  $\sigma_1$ , measurement-noise  $\sigma_2$ , distance per step  $v$ , headings  $h_k$ , magnetic measurements  $z_k$ , and magnetic-map interpolation function  $F(\text{Lat}, \text{Lon})$ .

- 1: Draw the initial particle set from the prior distribution:

$$x_0^{(j)} \sim \mathcal{N}(P_0, \sigma_1^0), \quad j = 1, \dots, N.$$

- 2: Assign equal initial particle weights:

$$w_0^{(j)} = \frac{1}{N}, \quad j = 1, \dots, N.$$

- 3: **for**  $k = 1, 2, \dots, K$  **do**

- 4:     Compute the dead reckoning change using distance, heading, and the Haversine formula  $\mathcal{H}$ :

$$\begin{aligned} \text{Lat}_k, \text{Lon}_k &= \mathcal{H}(\text{Lat}_{k-1}, \text{Lon}_{k-1}, v, h_k) \\ d_k &= \text{Lat}_k - \text{Lat}_{k-1}, \\ &\quad \text{Lon}_k - \text{Lon}_{k-1} \end{aligned}$$

- 5:     Propagate each particle using dead reckoning and process noise:

$$\begin{aligned} x_k^j &= x_{k-1}^j + d_k + \eta_k^j, \\ \eta_k^j &\sim \mathcal{N}(0, \sigma_1), \quad j = 1, \dots, N. \end{aligned}$$

- 6:     Predict the magnetic field value for each propagated particle:

$$\begin{aligned} \hat{z}_k^j &= F\left(\text{Lat}_{k|k-1}^j, \text{Lon}_{k|k-1}^j\right), \\ j &= 1, \dots, N. \end{aligned}$$


---

---

**Algorithm 2** Magnetic map matching particle filter implementation, continued

---

7:     Compute the measurement likelihood of each particle:

$$\ell_k^j = \frac{1}{\sqrt{2\pi} \sigma_m} \exp \left[ -\frac{\left( z_k - \hat{z}_k^j \right)^2}{2\sigma_m^2} \right],$$
$$j = 1, \dots, N.$$

8:     Update and normalise the particle weights:

$$\tilde{w}_k^j = w_{k-1}^j \ell_k^j,$$
$$w_k^j = \frac{\tilde{w}_k^j}{\sum_{r=1}^N \tilde{w}_k^r}, \quad j = 1, \dots, N.$$

9:     Estimate the corrected position as the weighted mean:

$$P'_k = \sum_{j=1}^N w_k^j x_k^j.$$

10:    Compute the effective number of particles:

$$N_{\text{eff}} = \frac{1}{\sum_{j=1}^N \left( w_k^j \right)^2}.$$

11:    **if**  $N_{\text{eff}}/N < 0.5$  **then**

12:       Draw a random offset for systematic resampling:

$$u_0 \sim \mathcal{U}\left(0, \frac{1}{N}\right).$$

13:       Define the systematic sample locations:

$$u_j = u_0 + \frac{j-1}{N},$$
$$j = 1, \dots, N.$$

14:       Select replacement particles by mapping each  $u_j$  through the cumulative distribution of the weights  $w_k^j$ .

15:       Reset the resampled particle weights:

$$w_k^j = \frac{1}{N}, \quad j = 1, \dots, N.$$

16:    **end if**

17: **end for**

---

## 5 Start point analysis

Table 1 provides additional error metrics for the results shown in Fig. 5 of the main manuscript. The value of 41.6 m in square brackets is the endpoint error of the dead reckoning path. The Cumulative Distribution Function (CDF) rows show the values below which 50% and 95% of the radial errors fall.

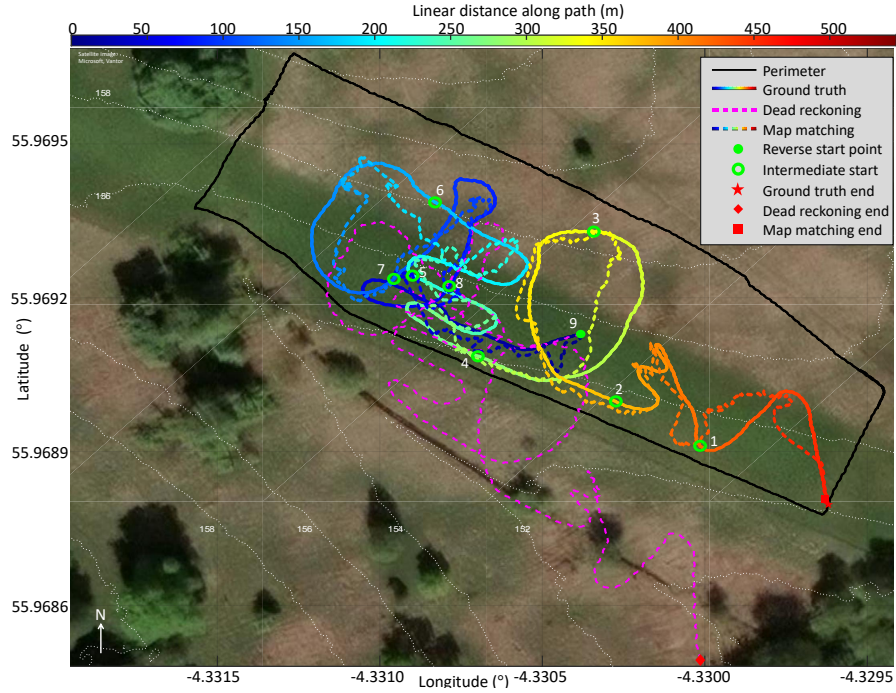
**Table 1:** Additional error metrics for the map matching results presented in the main manuscript.

| Metric                    | Value (m)  |
|---------------------------|------------|
| Mean radial error         | 2.8        |
| Median radial error       | 2.5        |
| Radial standard deviation | 1.8        |
| Endpoint error            | 6.5 [41.6] |
| CDF-50                    | 2.5        |
| CDF-95                    | 5.9        |

As a further analysis we examine the performance of the map matching approach over variable path lengths, variable directions, and from variable start points. This is achieved by taking the GNSS ground truth from the test path and selecting a total of 10 starting points comprising both ends of the path and 8 uniformly spaced intermediate points. From each start point the algorithm is run in two directions. “Forward” refers to a movement in the same direction as the physical trial; “reverse” refers to a scheme in which the reciprocal compass headings are used and the order of the OPM measurements reversed, i.e., using the real data to simulate movement backwards

**Table 2:** The error associated with the peak of the Beckmann fit for each direction from each starting point.

| Distance (m)        | Beckmann peak (m) |         |
|---------------------|-------------------|---------|
|                     | Forward           | Reverse |
| [Main manuscript] 0 | 2.2               | -       |
| [1] 61              | 2.2               | 1.3     |
| [2] 121             | 2.3               | 2.6     |
| [3] 182             | 2.4               | 1.8     |
| [4] 243             | 2.2               | 2.2     |
| [5] 303             | 2.2               | 1.9     |
| [6] 364             | 2.2               | 1.9     |
| [7] 424             | 1.8               | 2.6     |
| [8] 485             | 2.3               | 2.5     |
| [9] 545             | -                 | 3.1     |
| <b>Average</b>      | 2.2               | 2.2     |



**Fig. 5:** The reverse map matching path as well as the intermediate starting points. The figure is analogous to Fig. 4 in the main manuscript. The reversed path begins at the solid green marker (9). The green rings labelled 1 through 9 indicate the intermediate starting points corresponding to Table 2.

along the path. Each start point is treated as a hot start and the algorithm is run until it reaches the end of the path.

Table 2 and Fig. 5 summarise the results from multiple start points using the peak of the fitted Beckmann distribution. The distance column of Table 2 lists the distance along the path of the start points. The number in square brackets matches the labels and associated green rings in Fig. 5. The distance is measured from the South-Eastern start point on the boundary as shown in the main manuscript. The first row corresponds to the results in the main manuscript and has no reverse path since it is the end of the measured path.

Figure 5 is analogous to Fig. 4 in the main manuscript and shows the full dead reckoning and map matching path run in reverse, i.e., from start point 9 until the end of the path. Note that the dead reckoning path now diverges to the South of the test area rather than to the North. These results show that within our trial, the fitted radial error peak is robust to the chosen hot-start position, direction of travel, and path length.

## References

- [1] Li Q, Griffiths JG. Least squares ellipsoid specific fitting. In: Geometric modeling and processing, 2004. proceedings. IEEE; 2004. p. 335–340.
- [2] Yury.: Ellipsoid fit. Version 1.3.0.0; retrieved 17 June 2026. MATLAB Central File Exchange. Available from: <https://www.mathworks.com/matlabcentral/fileexchange/24693-ellipsoid-fit>.
- [3] Gordon NJ, Salmond DJ, Smith AF. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. In: IEE proceedings F (radar and signal processing). vol. 140. IET; 1993. p. 107–113.
- [4] Doucet A, Johansen A.: A tutorial on particle filtering and smoothing: fifteen years later. In The Oxford Handbook of Nonlinear Filtering. Oxford University Press.