

ПОБУДОВА БОРТОВОЇ СИСТЕМИ ВИЗНАЧЕННЯ КУТОВОЇ ОРІЄНТАЦІЇ НАНОСУПУТНИКА

Вступ

Реалізація алгоритмів керування кутовим рухом наносупутника методами простору станів потребує інформації про його повний фазовий вектор. Здебільшого складовими фазового вектору наносупутника є вектор абсолютної кутової швидкості та кватерніон орієнтації апарату, в цьому випадку фазовий вектор належатиме простору R^6 або R^7 . Якщо з метою підвищення точності керування повний фазовий вектор наносупутника розширюється компонентами збурюючого моменту, розмірність фазового вектора може зрости до 10. З цього випливає, що в умовах жорстких обмежень на доступні на борту енергетичні ресурси бортовий обчислювач наносупутника повинен забезпечувати оцінювання фазового вектора динамічної системи високого порядку з мінімальним енергоспоживанням.

Розв'язання даної проблеми потребує пошуку нових підходів до побудови бортової системи визначення кутової орієнтації (БСО) наносупутника. Не менш важливим є вибір алгоритму визначення орієнтації. Перспективним є застосування алгоритму розширеного фільтру Калмана, оскільки на його основі можна досягнути мінімальної дисперсії похибки оцінки фазового вектора наносупутника. При реалізації алгоритму розширеного фільтру Калмана на арифметиці з плаваючою або фіксованою комою виникає ряд проблем, пов'язаних з фіксованою точністю обчислень, що вносить додаткову методичну похибку в оцінку фазового вектора системи

Постановка задачі

Розглядається задача розробки мікропроцесорної архітектури БСО наносупутника. Обґрунтовується вибір апаратного забезпечення, розглядається алгоритм визначення кутової орієнтації та описуються проблеми реалізації алгоритму на арифметиці з фіксованою точністю, зокрема здійснюється оцінка методичної складової похибки визначення орієнтації, яка викликана обмеженою точністю обчислень у бортовому комп'ютері.

Математична модель алгоритму визначення кутової орієнтації

Задачею визначення кутової орієнтації наносупутника є задача визначення кватерніону орієнтації між зв'язаною і орбітальною системами координат (СК). Осі зв'язаної СК (ЗСК) фіксовані відносно корпусу наносупутника, її центр знаходиться в центрі мас супутника. Вимірювальні осі датчиків співпадають з осями ЗСК.

Орбітальна СК – це система координат, центр якої знаходиться в центрі мас супутника, вісь X_o спрямована за вектором лінійної швидкості космічного апарату, вісь Y_o спрямована до центру Землі, вісь Z_o доповнює орбітальну СК (ОСК) до правосторонньої прямокутної. У випадку, коли супутник знаходиться на сонячно-синхронній орбіті, вісь Y_o спрямована на Сонце.

Для оцінювання кутової орієнтації використовується алгоритм розширеного фільтру Калмана (АРФК) [1]. Для роботи алгоритму необхідно мати інформацію про абсолютну кутову швидкість наносупутника, вектор напрямку на Сонце та вектор магнітної індукції Землі, виражені в ЗСК, а також інформацію про еталонні вектор напрямку на Сонце та вектор магнітної індукції Землі, виражені в ОСК. Еталонний вектор магнітної індукції може бути розрахований в бортовому обчислювачі на основі математичних моделей магнітного поля Землі, таких як IGRF [2] чи WMF [3].

Розглядатимемо АРФК для визначення похибки оцінювання кутової орієнтації. В цьому випадку фазовий вектор запишеться як

$$X = [e_1 \quad e_2 \quad e_3]^T \quad (1)$$

Матриця стану системи виглядатиме наступним чином:

$$A = [-[\omega \times]] = \begin{bmatrix} 0 & \omega_z & -\omega_y \\ -\omega_z & 0 & \omega_x \\ \omega_y & -\omega_x & 0 \end{bmatrix}, \quad (2)$$

де $\omega_x, \omega_y, \omega_z$ – проекції абсолютної кутової швидкості на осі ЗСК, отримані з ДКШ. При переході до дискретної форми алгоритму потрібно обчислити матрицант системи за наступним виразом:

$$F = e^{A \times Ts} = I + A \times Ts + A \times \frac{Ts^2}{2!} + A \times \frac{Ts^3}{3!} + \dots, \quad (3)$$

де Ts – період дискретизації, зазвичай $Ts = 1\text{с}$.

Послідовність обчислень згідно з АРФК наведено у виразі (4):

$$\begin{aligned}
\bar{P}_k &= F \times P_{k-1} \times F^T + Q, \\
H &= \begin{bmatrix} 2[Bb \times] \\ 2[Sb \times] \end{bmatrix}, \\
K_k &= \bar{P}_k \times H^T \times (H \times \bar{P}_k \times H^T + R)^{-1}, \\
P_k &= (I_{3 \times 3} - K_k \times H) \times \bar{P}_k + K_k \times R \times K_k^T, \\
dX_k &= K_k \times \left(\begin{bmatrix} Bb_k \\ Sb_k \end{bmatrix} - \begin{bmatrix} \hat{R}_k \times Bo_k \\ \hat{R}_k \times So_k \end{bmatrix} \right), \\
dQ_k &= \begin{bmatrix} \sqrt{1 - dX_k^T \times dX_k} \\ \frac{dX_k}{4} \end{bmatrix}, \\
\hat{q}_k &= \hat{q}_{k-1} \otimes dQ_k.
\end{aligned} \tag{4}$$

У виразі (4) позначено:

Q – коваріаційна матриця шуму процесу;

R – коваріаційна матриця шуму вимірювання;

$\bar{P}_k$ – апіорна матриця коваріації похибки оцінювання фазового вектору;

H – матриця вимірювання;

Bb – вектор магнітної індукції Землі, виражений в ЗСК;

Sb – вектор напрямку на Сонце, виражений в ЗСК;

K_k – матриця посилення Калмана;

P_k – апостеріорна матриця коваріації похибки оцінювання фазового вектору;

$\hat{q}_k$ – оцінка кватерніону орієнтації наносупутника.

Символом $\otimes$ позначено операцію кватерніонного множення.

Архитектура БСО

На сучасному ринку обчислювальних пристроїв представлено широкий спектр мікроконтролерів, серед яких поставленій задачі найкраще відповідають МК з ядром ARM7 або ARM Cortex M4, оскільки такі мікроконтролери забезпечують високу обчислювальну потужність при невеликому споживанні електричної енергії. МК з ядром ARM Cortex M4 мають апаратний модуль для роботи з числами з плаваючою комою, тобто виконують математичні операції значно швидше, ніж МК з ядром ARM7, споживаючи таку саму кількість енергії.

Переважає більшість сучасних МК з ядром ARM7 або ARM Cortex M4 мають всі необхідні для побудови БСО периферійні пристрої, до яких

відносяться апаратні контролери інтерфейсів послідовної передачі даних (UART, SPI, I²C), аналого-цифрові перетворювачі, таймери. Архітектура БСО наведена на рис. 1. На рис. 1 позначено: ДКШ – датчик кутової швидкості, ДКС – датчик координат Сонця.



Рис. 1. Архітектура БСО

При розробці схеми поєднання датчиків первинної інформації та бортового обчислювача найкраще застосувати шину I²C, оскільки вона дозволяє поєднати до 127 пристроїв, в той час як інші послідовні інтерфейси (UART, SPI) дозволяють підключення тільки одного пристрою.

Моделювання роботи БСО

Для перевірки працездатності розробленої БСО та для оцінки її точності було проведено програмно-апаратне моделювання. З метою імітації ДКС при виконанні моделювання було використано акселерометр ADXL345. В якості бортового обчислювача було використано МК з ядром ARM7 – AT91SAM7S256 [4], подібний до МК, використаного в комерційно доступному бортовому комп'ютері NanoMind [5].

При розробці програмного забезпечення необхідно врахувати низку особливостей реалізації алгоритму на арифметиці з фіксованою точністю. Внаслідок обмеженості діапазону значень змінних типу double або float, в процесі обчислень виконуються округлення, які призводять до появи похибки в оцінці кутової орієнтації. Найбільш чутливими до таких округлень є операції з погано обумовленими матрицями, а також операції множення та ділення, в яких хоча б один з операндів близький до нуля за абсолютною величиною.

Для зменшення впливу похибок обчислень на кінцевий результат, необхідно правильно обрати тип даних для представлення змінних. Застосування типу `double` потребує більш потужного обчислювача, однак при цьому результат буде точніший, ніж при використанні типу `float`. В процесі моделювання встановлено, що при використанні змінних типу `float` в системі деякий час після запуску має місце втрата числової стійкості, що означає неможливість використання даних про кутову орієнтацію для керування рухом супутника в цей час.

Для обчислення матрицанту системи згідно з виразом (3), потрібно обчислити суму ряду $I + A \times Ts + A \times \frac{Ts^2}{2!} + A \times \frac{Ts^3}{3!} + \dots A \times \frac{Ts^n}{n!}$, попередньо визначивши кількість членів, які потрібно врахувати, оскільки при їх недостатній кількості можлива втрата точності в моменти, коли супутник виконує поворот, тобто модуль вектора його абсолютної кутової швидкості не дорівнює нулеві. На рис. 2 та рис. 3 наведено похибки визначення кутів орієнтації при різних кількостях врахованих членів ряду. Як бачимо, при $n \geq 2$ точність визначення кутової орієнтації суттєво не змінюється рис. 4 та рис. 5.

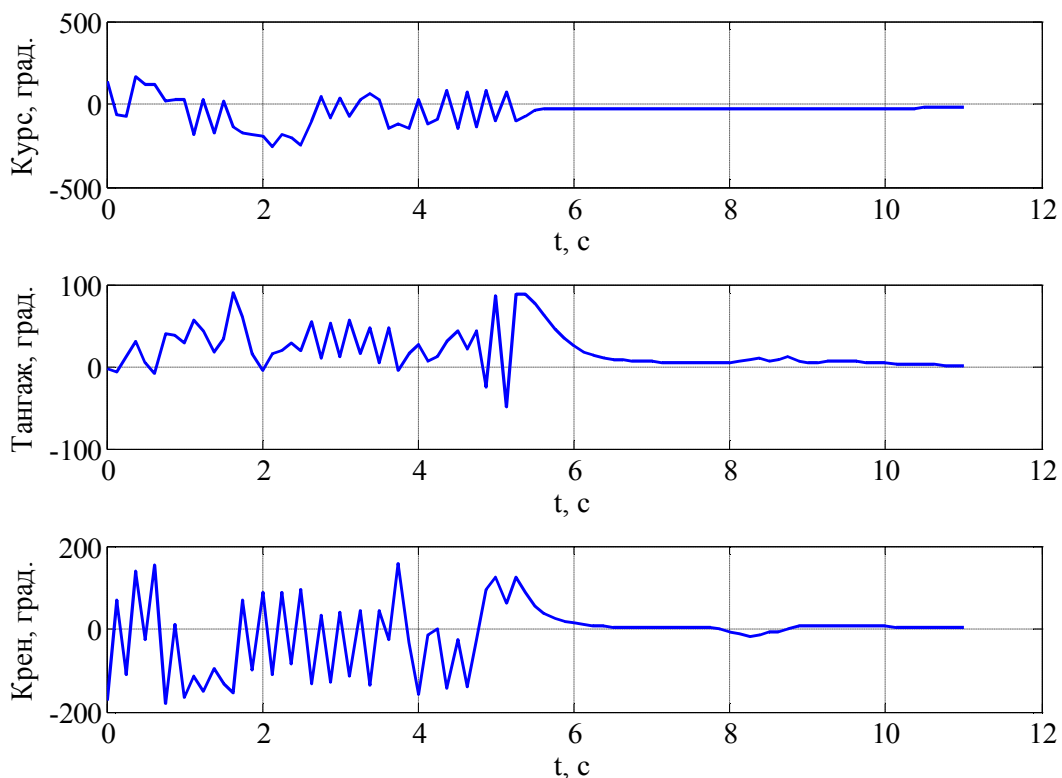


Рис. 2. Похибка визначення кутів орієнтації при використанні типу даних `float`

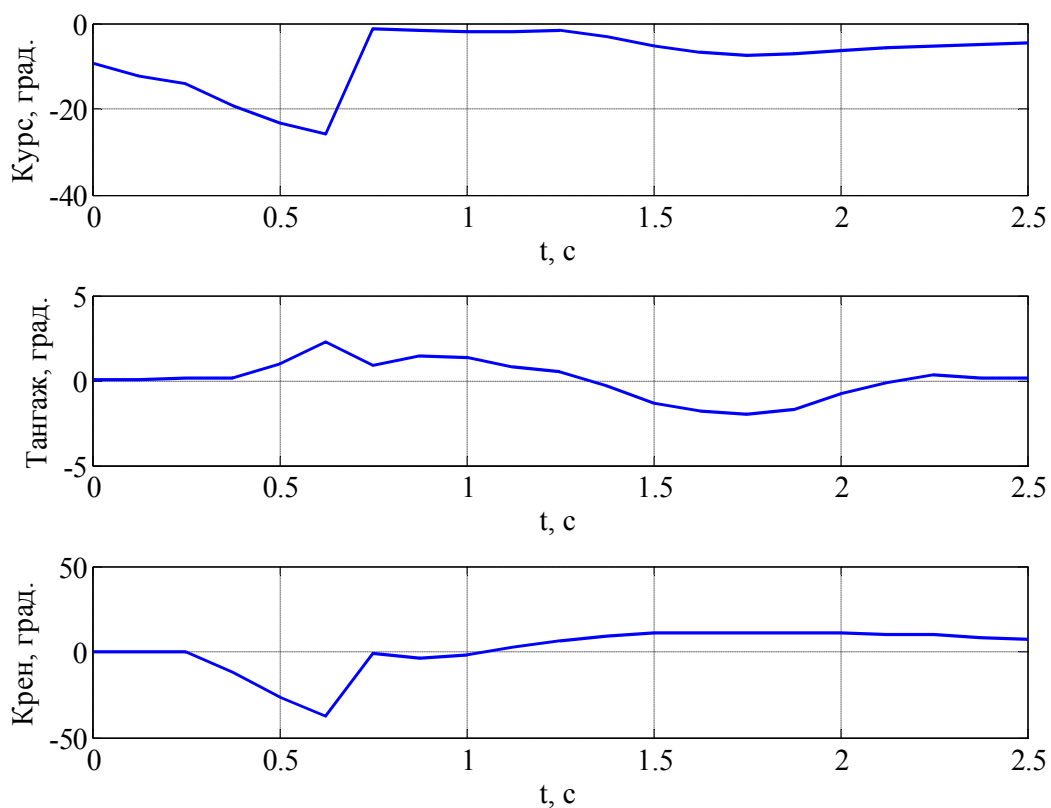


Рис. 3. Похибка визначення кутів орієнтації при використанні типу даних double

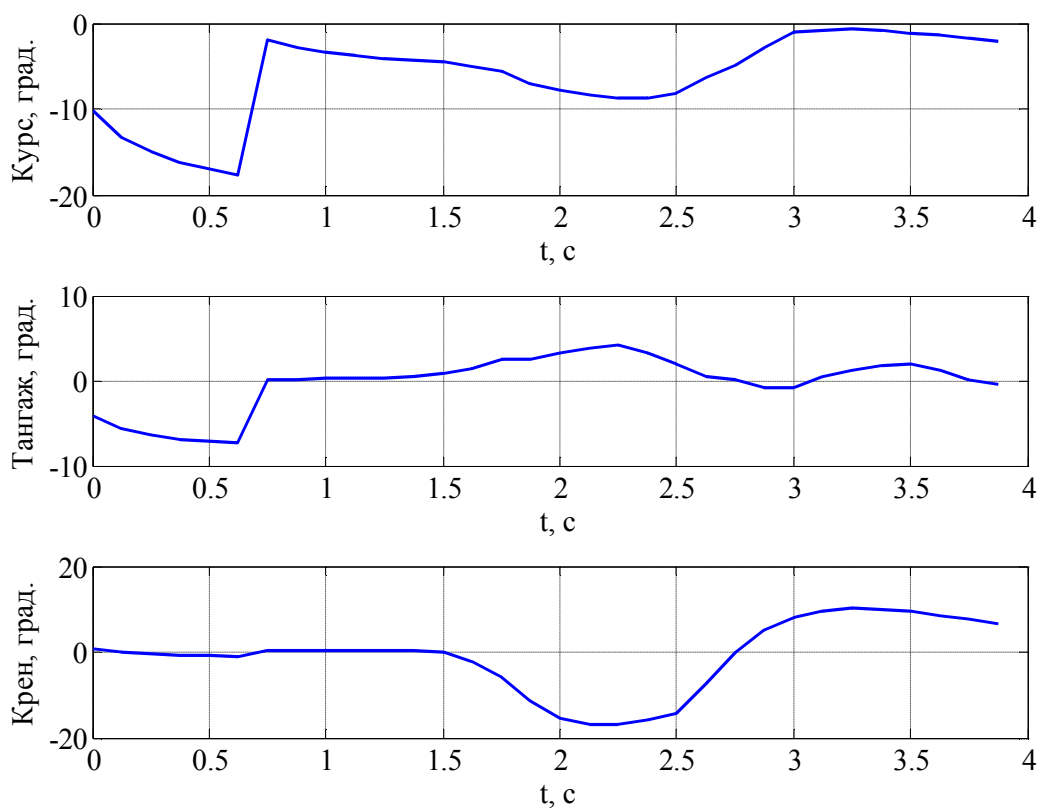


Рис. 4. Похибка визначення кутів орієнтації при $n = 2$

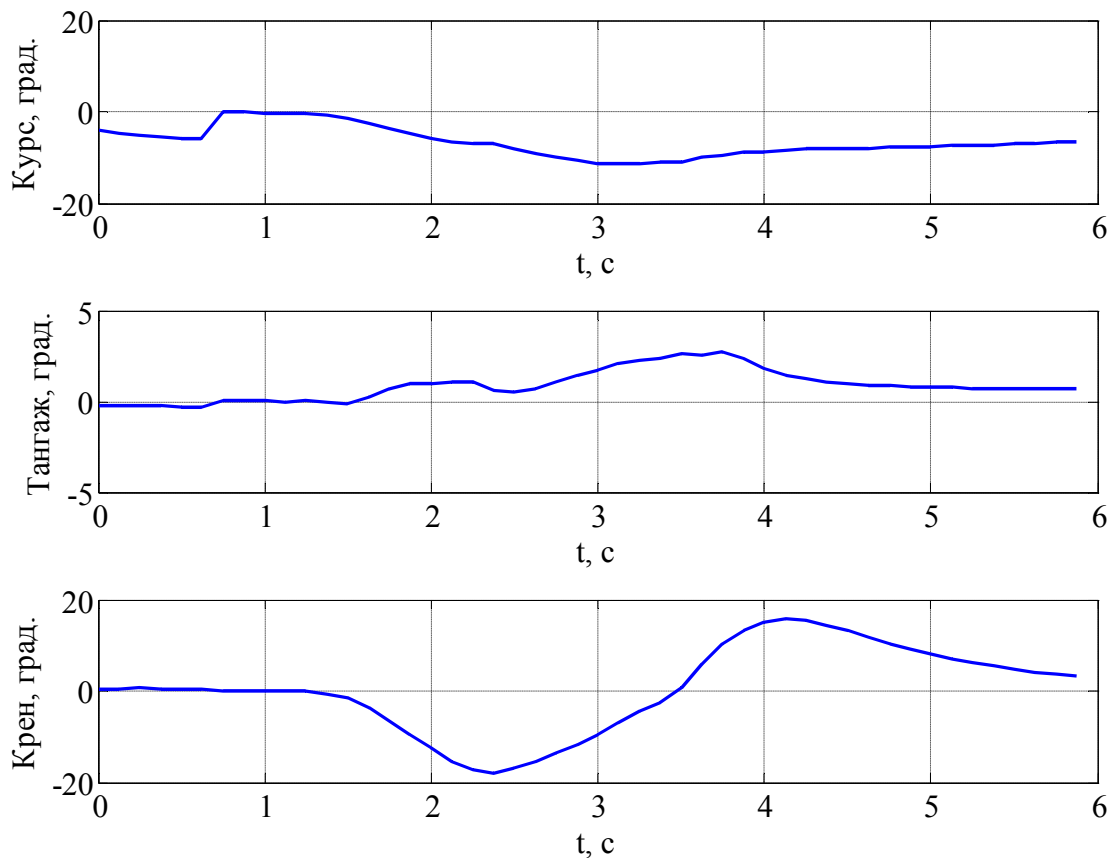


Рис. 5. Похибка визначення кутів орієнтації при $n = 3$

Висновки

Встановлено, що при використанні в програмному забезпеченні змінних з плаваючою комою одинарної точності, алгоритм визначення кутової орієнтації є нестійким в початковий період і заспокоюється на протязі приблизно 1 хв. Використання змінних подвійної точності дозволяє уникнути цього явища, однак несуттєво покращує точність при роботі алгоритму після початкового заспокоєння.

Встановлено, що для обчислення матричної експоненти достатньо обмежитись першими двома членами ряду Тейлора, оскільки збільшення кількості членів, що враховуються, не призводить до суттєвого підвищення точності визначення кутової орієнтації.

Список використаної літератури

1. Marco Schmidt, Karthik Ravandoor, Oliver Kurz, Stephan Busch and Klaus Schilling. Attitude Determination for the Pico-Satellite UWE-2. In The International Federation of Automatic Control papers archive, 2008.
2. Jeremy J. Du Croz. Stability of methods of matrix inversion. In IMA Journal of Numerical analysis, 1992.

3. Jeremy Davis. Mathematical modeling of the Earth magnetic field. Virginia Tech, Blacksburg, VA 24061, 2004.
4. Математична модель магнітного поля Землі WMM. Режим доступу: <http://www.ngdc.noaa.gov/geomag/WMM/back.shtml>.
5. Паспорт на мікроконтролер AT91SAM7S256. Режим доступу: <http://www.atmel.com/images/doc6175.pdf>.
6. Паспорт на бортовий комп'ютер Nanomind. Режим доступу: <http://gomspace.com/index.php?p=products-a712c>.