

MODIFIKASI PERSAMAAN QUATERNION PADA ALGORITMA INS UNTUK APLIKASI ROKET

Romi Wiryadinata^{*)}, Wahyu Widada^{**)}

^{*)} Jurusan Teknik Elektro, Fakultas Teknik, Univeristas Sultan Ageng Tirtayasa
e-mail: romi_wiryadinata@yahoo.com

^{**)} Peneliti Bidang Telemetry dan Muatan Roket, LAPAN
email: w_widada@yahoo.com

ABSTRACT

INS algorithm is used to transform the body frame to the inertial frame of the rocket trajectory. The motion of rocket rotation is different from the air plane. The rocket has a faster motion and need a high speed computation. The pitch angle (θ) which is being greater than 90° , there will be a problem and it needs the Quaternion equation to solve the problem. In this study, the Quaternion parameters are always normalized with the same value at the time of initial initialization. The results of this study indicate that the calculation of body frame and inertial frames have similar results and showed accuracy between 0.2 - 1 meter. In the process algorithms in real-time INS will require a high-speed microprocessors to speed up the computing time on an embedded system payload.

Keywords: *Inertial Navigation System, Inertial Measurement Unit, Quaternion, Rocket*

ABSTRAK

Algoritma INS digunakan untuk mentransformasikan *body frame* ke *inertial frame* pada trayektori. Gerak rotasi roket berbeda dengan pesawat, gerakan roket sangat ekstrim dan membutuhkan kecepatan komputasi yang tinggi. Sudut pitch (θ) lebih dari $\pm 90^\circ$ akan menimbulkan masalah, sehingga persamaan *Quaternion* sebagai matrik transformasi untuk algoritma INS diperlukan untuk mengatasi masalah tersebut. Pada penelitian ini, parameter *Quaternion* selalu dinormalisasi dengan nilai yang sama dengan pada saat inisialiasi awal. Hasil penelitian ini menunjukkan bahwa perhitungan *body frame* dan *inertial frame* memiliki hasil yang mirip dan menunjukkan akurasi antara 0.2 – 1 meter. Pada proses algoritma INS secara *real-time* akan membutuhkan mikroprosesor berkecepatan tinggi untuk mempercepat waktu komputasi pada sistem *embedded payload*.

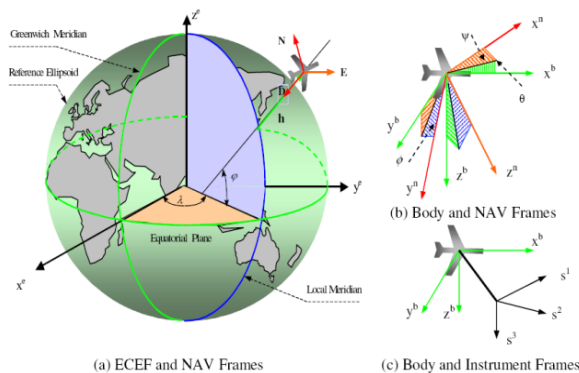
Kata Kunci: *Inertial Navigation System, Inertial Measurement Unit, Quaternion, Roket*

1 PENDAHULUAN

Sistem navigasi yang telah lama dikenal manusia untuk membantu menemukan arah terus dikembangkan untuk berbagai macam keperluan. Manusia sebelum mengenal kompas menggunakan rasi bintang sebagai acuan untuk menentukan arah utara dan selatan, dan matahari sebagai acuan untuk menentukan arah barat dan timur. Semakin berkembangnya teknologi kompas dikembangkanlah kompas digital baik yang berbasis magnet maupun kompas yang berbasis sudut.

Kompas cocok digunakan sebagai penentu arah navigasi yang sejajar dengan permukaan Bumi (ketinggian diabaikan) dan hanya membutuhkan berdasarkan mata angin, sehingga dikembangkanlah sistem navigasi yang lebih lengkap hingga keluarannya tidak hanya arah mata angin, tetapi koordinat bumi bahkan sampai ketinggiannya. (Flenniken, 2005) mengemukakan pendapatnya bahwa pergerakan dari benda (utara, timur, arah ke atas) akan selalu mencari acuan terhadap Bumi, benda, navigasi, geografis, maupun inersia.

Ada beberapa transformasi atau tata acuan koordinat (*reference frame*) dan rotasi seperti *Earth Centred Earth Fix* (ECEF), *Earth Centred Inertial* (ECI), *Local Geodetic Vertical* (LGV), *Wander Azimuth Frame* (WAF), dan *body frame* (Walchko, 2002) yang secara umum dapat digambarkan dengan peta dasar sistem navigasi yang terdapat (Gambar 1-1) di bawah ini. Pada penelitian ini digunakan tata acuan koordinat *inertial* atau navigasi dan difokuskan pada algoritma untuk *strapdown Inertial Navigation System* (INS) pada roket kendali Lembaga Penerbangan dan Antariksa Nasional (LAPAN) dengan menggunakan *Quaternion* untuk mengetahui rotasi benda pada sistem koordinat.



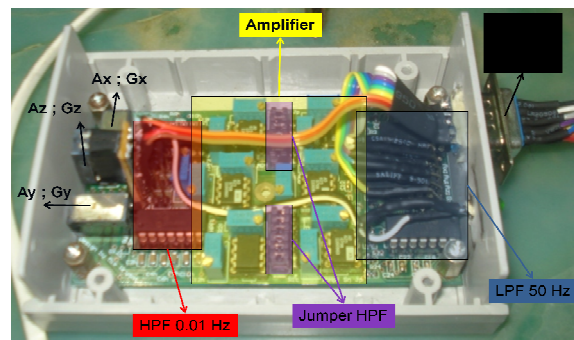
Gambar 1-1: Dasar sistem navigasi (Jia, 2004)

Pada tulisan ini akan dijelaskan bagaimana algoritma INS digunakan pada roket yang bergerak tidak lazim dibandingkan dengan benda terbang lainnya. Kecepatan roket yang mencapai 530 meter/detik membuat algoritma dan metode *Quaternion* yang digunakan dimodifikasi terlebih dahulu menyesuaikan kebutuhan untuk objek roket. Sehingga algoritma *strapdown* INS yang digunakan juga tidak sama dengan algoritma yang umum digunakan.

Normalisasi metode *Quaternion* yang digunakan pada algoritma INS diubah sedemikian rupa hingga hasilnya sama dengan metode *Quaternion* yang digunakan tetapi berbeda dengan metode pada umumnya. Begitu juga

algoritma INS yang digunakan biasanya tidak membutuhkan *buffering*, tetapi pada algoritma INS untuk roket ini modifikasi-modifikasi dilakukan hingga terbukti bahwa perubahan referensi sumbu *body* menjadi referensi *inertial* memiliki pola yang sama tetapi dengan nilai yang berbeda walaupun dengan selisih nilai yang sedikit karena adanya parameter-parameter dan faktor-faktor lain dari unit pengukuran inersia atau dikenal dengan *Inertial Measurement Unit* (IMU).

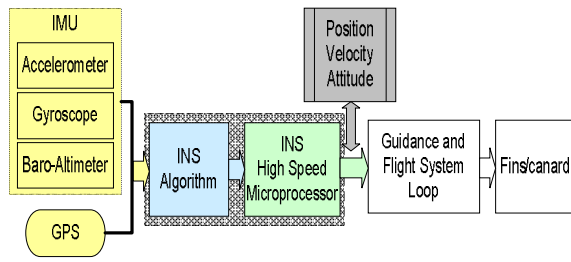
IMU terdiri dari 3 axis sensor *accelerometer* dan 3 axis sensor *gyroscope*. *Accelerometer* memiliki keluaran percepatan pada 3 arah sumbu yang masing-masing diberi notasi ax sebagai percepatan pada arah sumbu x, ay percepatan pada arah sumbu y, dan az merupakan percepatan pada arah sumbu z, sedangkan *gyroscope* memiliki keluaran berupa kecepatan sudut dari arah sumbu x yang nantinya akan menjadi sudut ϕ (*roll*), dari sumbu y nantinya menjadi sudut θ (*pitch*), dan sumbu z nantinya menjadi sudut ψ (*yaw*). Berikut ini (Gambar 1-2) adalah salah satu jenis IMU yang dikembangkan pada penelitian ini.



Gambar 1-2: Inertial Measurement Unit

2 MODIFIKASI ALGORITMA INS

Variasi dari jenis dan penggunaan IMU serta INS akan sangat berpengaruh terhadap *performance* jenis roket, sehingga kesalahan dapat lebih ditekan seminimal mungkin, berikut ini (Gambar 2-1) adalah diagram blok algoritma INS lengkap dengan sistem kendalinya.



Gambar 2-1: Diagram Blok INS dan sistem kendali roket (Wiryadinata, 2009)

Dari (Gambar 2-1) di atas, algoritma yang dikembangkan pada penelitian ini nantinya akan diimplementasikan ke dalam sebuah mikroprosesor berkecepatan tinggi mengingat instruksi yang banyak dan kecepatan roket yang tinggi akan membutuhkan komputasi yang handal dan *hardware* yang handal juga, tetapi pada penelitian ini tidak membahas sistem mikroprosesor, *guidance, flight*, dan bagian *fins/canard*.

Algoritma INS yang dikembangkan, ada beberapa modifikasi salah satunya seperti yang dilakukan (Kumar, 2004) menurunkan persamaan gerak Euler dan mengganti parameter Quaternion dengan variabel Euler, lihat pada (persamaan 2-1 sampai 2-4) berikut. Parameter Euler ($e_0, e_1, e_2,$ dan e_3) pada persamaan tersebut digantikan dengan parameter Quaternion ($q_0, q_1, q_2,$ dan q_3) sehingga setiap persamaannya akan digunakan untuk perhitungan, sedangkan persamaan Euler hanya digunakan sebagai langkah awal pada inialisasi penentuan parameter.

$$\dot{e}_0 = -\frac{1}{2}(e_1p + e_2q + e_3r) \quad (2-1)$$

$$\dot{e}_1 = \frac{1}{2}(e_0p + e_2r - e_3q) \quad (2-2)$$

$$\dot{e}_2 = \frac{1}{2}(e_0q + e_3p - e_1r) \quad (2-3)$$

$$\dot{e}_3 = \frac{1}{2}(e_0r + e_1q - e_2p) \quad (2-4)$$

Setelah nilai parameter Euler diperoleh kemudian parameter tersebut diintegrasikan seperti terdapat pada (persamaan 2-5 sampai 2-8) agar diperoleh nilai parameter Euler untuk selanjutnya disubstitusikan pada (persamaan 2-10 sampai 2-12).

$$e_{0(n)} = e_{0(n-1)} + \dot{e}_0 dt \quad (2-5)$$

$$e_{1(n)} = e_{1(n-1)} + \dot{e}_1 dt \quad (2-6)$$

$$e_{2(n)} = e_{2(n-1)} + \dot{e}_2 dt \quad (2-7)$$

$$e_{3(n)} = e_{3(n-1)} + \dot{e}_3 dt \quad (2-8)$$

Bentuk lain (persamaan 2-5 sampai 2-8) bisa juga dituliskan dalam bentuk matematis seperti pada (persamaan 2-9) berikut ini.

$$\left. \begin{aligned} e_0 &= \int \dot{e}_0 dt \\ e_1 &= \int \dot{e}_1 dt \\ e_2 &= \int \dot{e}_2 dt \\ e_3 &= \int \dot{e}_3 dt \end{aligned} \right\} \quad (2-9)$$

Nilai sudut *roll* (φ), *pitch* (θ), dan *yaw* (ψ) ditentukan oleh (persamaan 2-10 sampai 2-12). Keluaran dari ketiga persamaan tersebut adalah berupa sudut dalam tata acuan *inertial*.

$$\Delta\theta = -\text{asin}(2(q_1q_3 - q_0q_2)) \quad (2-10)$$

$$\Delta\varphi = \text{atan2}(2(q_2q_3 + q_0q_1), (1 - 2(q_1^2 + q_2^2))) \quad (2-11)$$

$$\Delta\psi = \text{atan2}(2(q_1q_2 + q_0q_3), (1 - 2(q_2^2 + q_3^2))) \quad (2-12)$$

Normalisasi parameter Quaternion dilakukan seperti pada (persamaan 2-13) tetapi dengan menetapkan nilai-nilai parameternya menjadi seperti pada (persamaan 2-14), normalisasi ini yang membuat berbeda dengan algoritma INS pada umumnya. Nilai parameter yang seharusnya di-*update* dengan mengkondisikan beberapa parameter sebelumnya pada algoritma ini nilai tersebut ditetapkan menjadi sebuah konstanta yang akan digunakan untuk meng-*update* nilai parameter Quaternion.

$$e_0^2 + e_1^2 + e_2^2 + e_3^2 = 1 \quad (2-13)$$

$$\left. \begin{aligned} e_0 &= 1 \\ e_1 &= e_2 = e_3 = 0 \end{aligned} \right\} \quad (2-14)$$

Sehingga bukan perubahan nilai parameter yang dihitung pada data $n+1$ tetapi berupa perubahan sudut yang ditambahkan dengan nilai pada sudut sebelumnya (persamaan 2-15 sampai 2-17). Persamaan ini juga merupakan persamaan yang membuat berbeda antara algoritma untuk objek pesawat

algoritma yang digunakan pada roket. Nilai *roll* (φ), *pitch* (θ), dan *yaw* (ψ) sebelumnya diubah terlebih dahulu kedalam bentuk radian dengan mengalikan dengan $180/\pi$.

$$\Phi_{(n)} = \Delta\varphi_{(n)} + \Phi_{(n-1)} \quad (2-15)$$

$$\Theta_{(n)} = \Delta\theta_{(n)} + \Theta_{(n-1)} \quad (2-16)$$

$$\Psi_{(n)} = \Delta\psi_{(n)} + \Psi_{(n-1)} \quad (2-17)$$

Nilai *roll* (φ), *pitch* (θ), dan *yaw* (ψ) ini nantinya akan digunakan pada persamaan yang dibutuhkan pada saat memproses *accelerometer*, tetapi sebelum ke persamaan berikutnya dan merupakan nilai sudut rotasi dari sumbu inertial yang digunakan. Nilai sudut *roll*, *pitch*, dan *yaw* diubah kembali ke dalam bentuk sudut dengan mengalikan dengan $\pi/180$.

Modifikasi yang dilakukan pada algoritma INS berada pada (persamaan 2-14 sampai 2-17), dengan menjumlah perubahan sudut saat ini dengan sudut sebelumnya maka gerak rotasi roket yang cepat dan terlebih dengan menggunakan parameter *Quaternion* maka gerakan ekstrim roket seperti *roll* yang sangat cepat (lebih dari 300 deg/sec dan sudut pitch 90°) dapat diatasi. Selain itu untuk gerakan yang cepat dengan frekuensi tinggi, sudut yang dihasilkan jika dengan algoritma biasa hanya akan mencapai 45°.

$$ax_{corr} = a_x - g \sin\theta \quad (2-18)$$

$$ay_{corr} = a_y + g \cos\theta \sin\varphi \quad (2-19)$$

$$az_{corr} = a_z + g \cos\theta \cos\varphi \quad (2-20)$$

Sampai tahap ini proses perhitungan dari keluaran sensor *gyroscope* sudah dapat diamati hasilnya dan langkah berikutnya adalah proses perhitungan untuk keluaran dari sensor *accelerometer* yang mengkombinasikan juga dengan keluaran dari sensor *gyroscope*. Langkah pertama yang dilakukan berikutnya adalah melakukan koreksi gravitasi (persamaan 2-18 sampai 2-20) yang berfungsi untuk menghilangkan pengaruh gravitasi dari *accelerometer* sehingga nilai percepatan

tidak lagi dipengaruhi adanya gaya gravitasi. Persamaan tersebut dimodifikasi dari persamaan untuk menghitung kecepatan seperti terdapat pada (persamaan 2-21 sampai 2-23).

$$\dot{U} = ax_{corr} + Vr - Wq + g \sin\theta \quad (2-21)$$

$$\dot{V} = ax_{corr} - Ur + Wp - g \cos\theta \sin\varphi \quad (2-22)$$

$$\dot{W} = ax_{corr} + Uq - Vp - g \cos\theta \cos\varphi \quad (2-23)$$

(Persamaan 2-21 sampai 2-23) adalah persamaan untuk menghitung kecepatan. \dot{U} sebagai kecepatan pada arah sumbu x; \dot{V} pada arah sumbu y; dan \dot{W} pada arah sumbu z, perlu di-integral untuk memperoleh kecepatan (U, V , dan W) pada sumbu referensi *inertial*. Proses integral dilakukan seperti pada (persamaan 2-24 sampai 2-26) atau secara matematis dapat dituliskan pada (persamaan 2-27) dan berlaku juga untuk nilai V dan W .

$$U_{(n)} = U_{dot(n)} + U_{(n-1)} * dt \quad (2-24)$$

$$V_{(n)} = V_{dot(n)} + V_{(n-1)} * dt \quad (2-25)$$

$$W_{(n)} = W_{dot(n)} + W_{(n-1)} * dt \quad (2-26)$$

$$\left. \begin{aligned} U &= \int \dot{U} dt \\ V &= \int \dot{V} dt \\ W &= \int \dot{W} dt \end{aligned} \right\} \quad (2-27)$$

Setelah nilai percepatan dan kecepatan diketahui, kemudian jarak dapat diperoleh dengan mengalikan nilai kecepatan dengan matriks transformasi *Direct Cosine Matrix* (DCM) yang juga dengan menggunakan parameter *Quaternion* seperti terdapat pada (persamaan 2-28)

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = \begin{bmatrix} V_N \\ V_E \\ V_D \end{bmatrix} = DCM^T \begin{bmatrix} U \\ V \\ W \end{bmatrix} \quad (2-28)$$

Hasil perkalian tersebut masih dalam bentuk kecepatan dalam arah koordinat Bumi, V_N adalah kecepatan ke-arah utara, V_E kecepatan ke-arah timur, dan V_D kecepatan ke-arah bawah searah dengan gravitasi Bumi. Karena nilai jarak diperoleh dalam satuan meter maka kecepatan-kecepatan tersebut perlu diintegral (persamaan 2-29 sampai

2-31) yang dapat dituliskan dalam persamaan matematis seperti terdapat pada (persamaan 2-32).

$$X_{(n)} = Xdot_{(n)} + X_{(n-1)} * dt \quad (2-29)$$

$$Y_{(n)} = Ydot_{(n)} + Y_{(n-1)} * dt \quad (2-30)$$

$$Z_{(n)} = Zdot_{(n)} + Z_{(n-1)} * dt \quad (2-31)$$

$$\left. \begin{aligned} X &= \int \dot{X} dt \\ Y &= \int \dot{Y} dt \\ Z &= \int \dot{Z} dt \end{aligned} \right\} \quad (2-32)$$

Persamaan gerak rotasi untuk menghitung keluaran dari *gyroscope* yang berupa kecepatan sudut (*deg/sec*) sampai dengan menghasilkan rotasi dalam bentuk sudut (*degree*) terdapat pada (persamaan 2-1) sampai dengan (persamaan 2-17), sedangkan untuk mengetahui persamaan gerak translasi dari *accelerometer* terdapat pada (persamaan 2-18) sampai dengan (persamaan 2-32) walaupun secara fisik sensor tersebut dipisah, tetapi untuk mengetahui pergerakan dalam ruang 3D dan dalam tata acuan navigasi atau inersia, kedua keluaran sensor tersebut memiliki kaitan yang erat dan saling berhubungan.

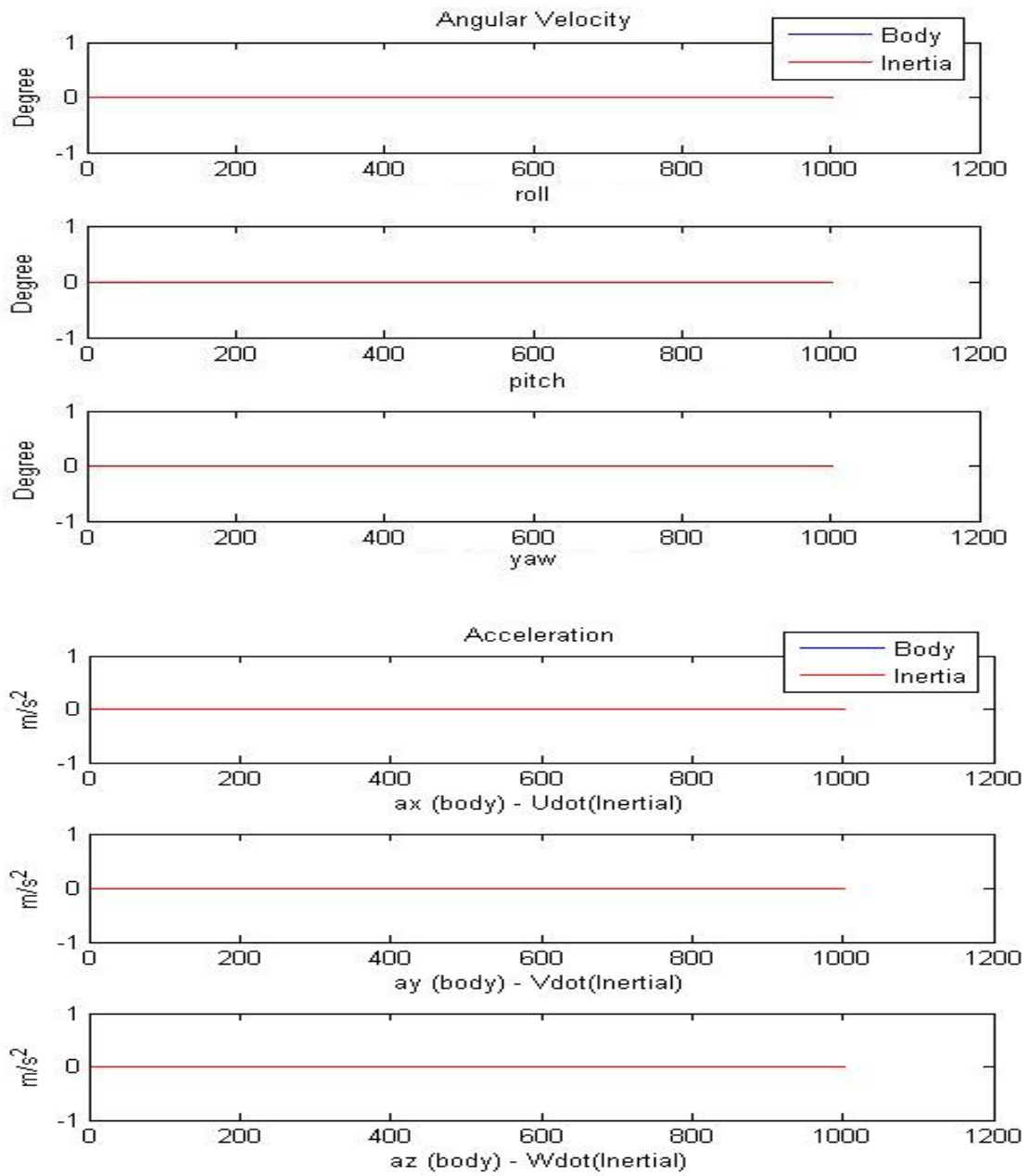
3 HASIL PENELITIAN

Pada penelitian ini, ujicoba dilakukan dengan mengganti parameter-parameter yang merupakan data keluaran dari IMU. Modifikasi dilakukan beberapa kali hingga *output* dari algoritma INS antara data dari sensor dan keluaran algoritma INS memiliki pola yang sama, tetapi tanpa mengubah tujuan dan algoritma dasar yang ada. Ujicoba pertama kali dilakukan dengan membandingkan data saat objek benda

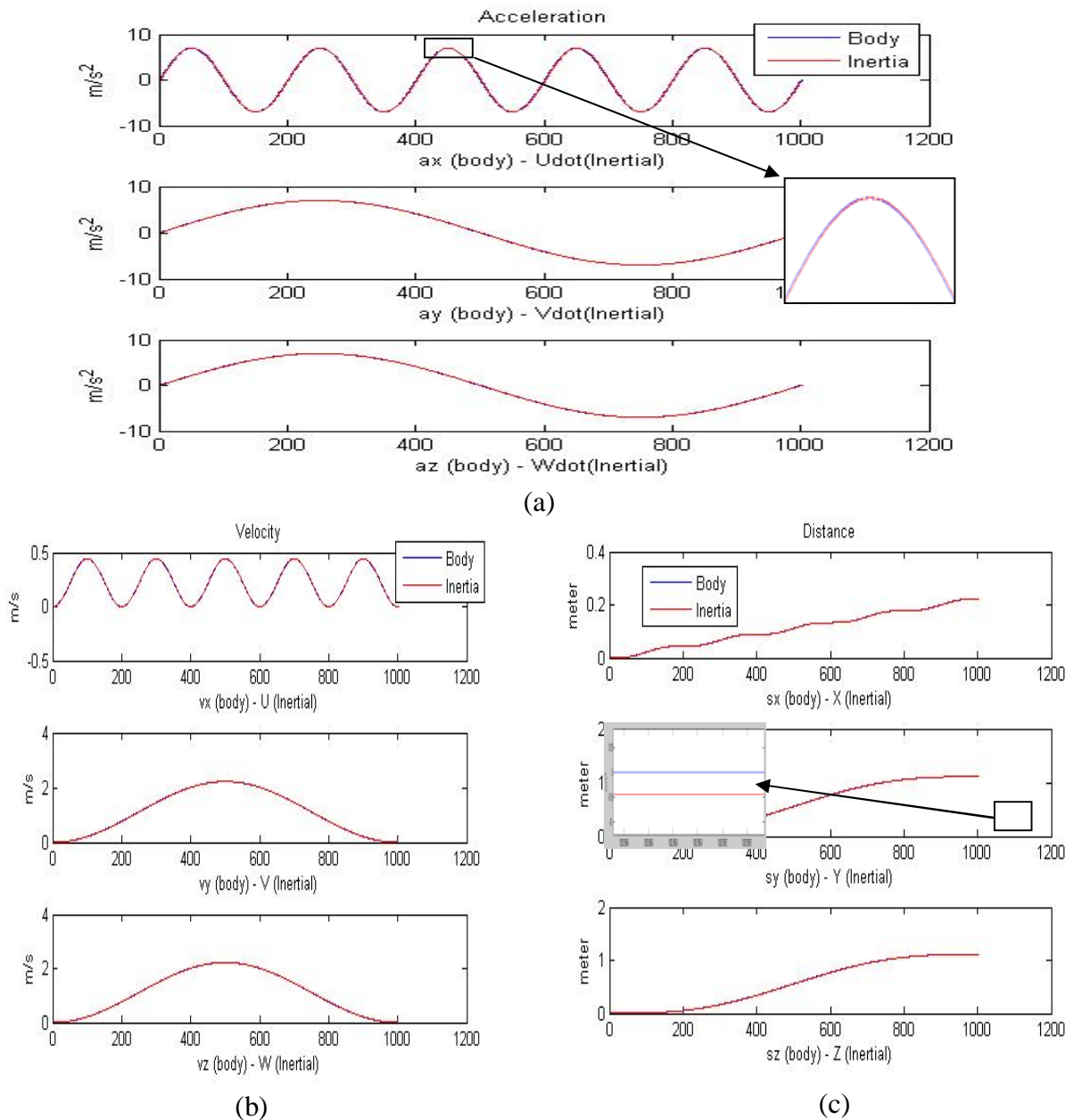
dalam keadaan diam, kemudian satu persatu setiap sumbu diubah (digerakan) dengan memberikan nilai sederhana berupa gelombang sinus yang menandakan adanya pergerakan/perubahan hingga kembali pada kondisi semula. Berikut ini (Gambar 3-1) adalah keluaran masing-masing sensor setiap sumbu yang tidak digerakan atau pada saat keadaan diam.

Percobaan berikutnya adalah mengamati perubahan yang terjadi pada *accelerometer* ketika digerakan tetapi *gyroscope* dalam keadaan diam. *Accelerometer* sumbu x diberikan perubahan data (pergerakan) 5 Hz sedangkan sumbu y dan z diberikan data perubahan sebesar 1 Hz. Dapat dilihat pada (Gambar 3-2) pola pergerakan antara keluaran sensor dengan hasil keluaran algoritma INS memiliki pola yang sama. Pada beberapa titik sampel pengamatan, grafik diperbesar untuk melihat dan membuktikan adanya perbedaan antara data *input (body)* dan data *output (inertial)*, karena *gyroscope* tidak ada perubahan (tidak ada rotasi dari objek benda) maka hasil integral dari percepatan dan kecepatan tidak terjadi perubahan yang terlalu signifikan.

Percobaan ini juga dilakukan pada *gyroscope*, dengan memberi pergerakan *gyroscope* tetapi dengan *accelerometer* yang tidak bergerak dalam keadaan diam. Setelah grafik pengamatan yang diamati memiliki pola sama, ini berarti bahwa INS algoritma telah terbukti sesuai dan dapat dilakukan pengamatan untuk data yang lain dan mengamati setiap perubahannya.



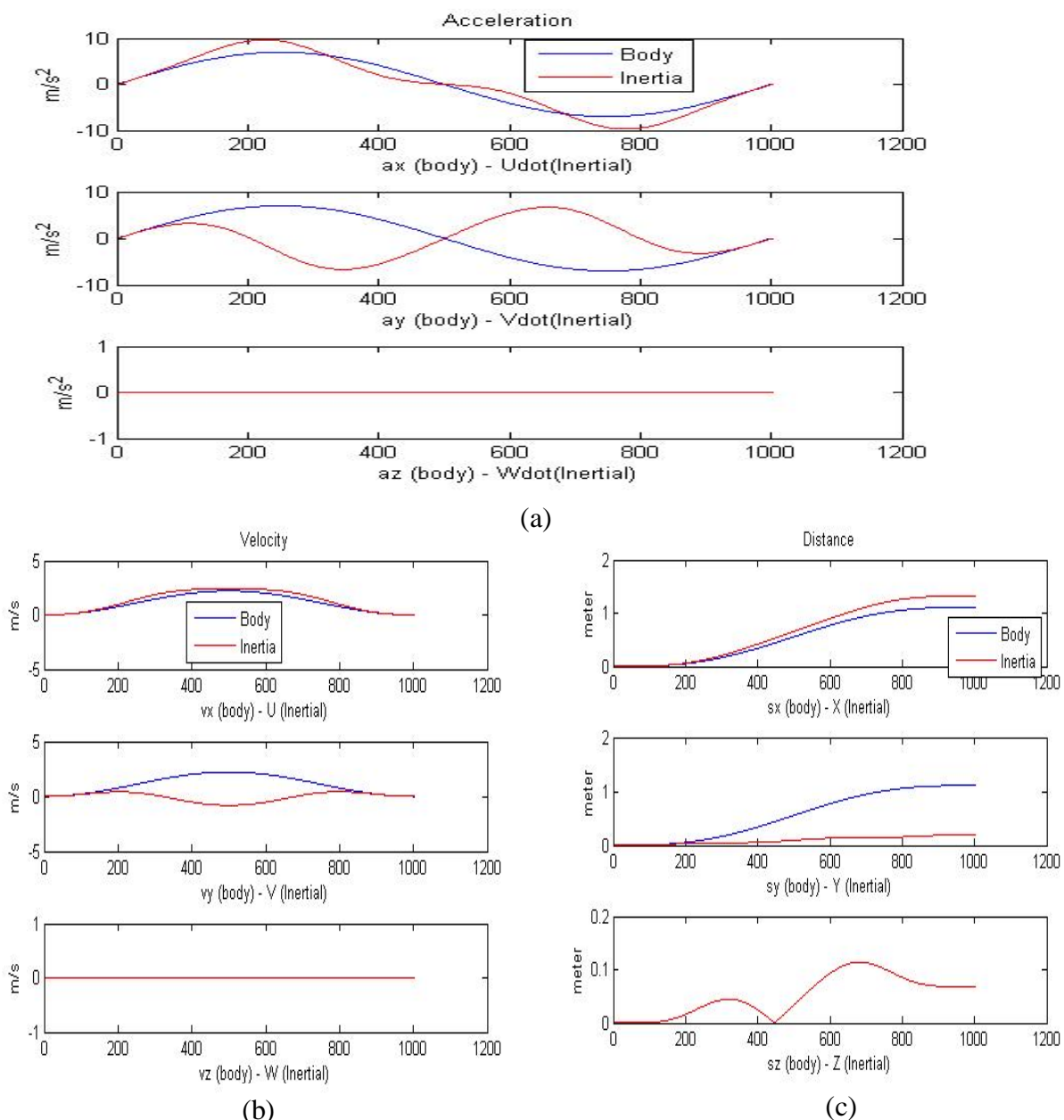
Gambar 3-1: Kecepatan sudut *gyroscope* (atas) dan percepatan *accelerometer* (bawah) pada saat keadaan diam



Gambar 3-2: Accelerometer sumbu x diberikan frekuensi 5 Hz dan untuk sumbu y dan z sebesar 1 Hz, (a) percepatan, (b) kecepatan, dan (c) jarak

Pengamatan berikutnya adalah percobaan dengan mensimulasikan sensor *gyroscope* dengan menggerakkan *gyroscope* sumbu z, yang berarti bahwa dalam keadaan sebenarnya objek benda bergerak dalam keadaan bidang datar sebesar 1 Hz (Gambar 3-3). Percobaan seperti ini adalah untuk menguji algoritma INS pada kendaraan darat yang hanya bergerak pada sumbu z (*yaw*), tanpa ada perubahan atau pergerakan terhadap sumbu x (*roll*) dan y (*pitch*), diasumsikan bahwa rotasi terjadi pada sumbu z bukan pada sumbu x (sesuai pergerakan kendaraan).

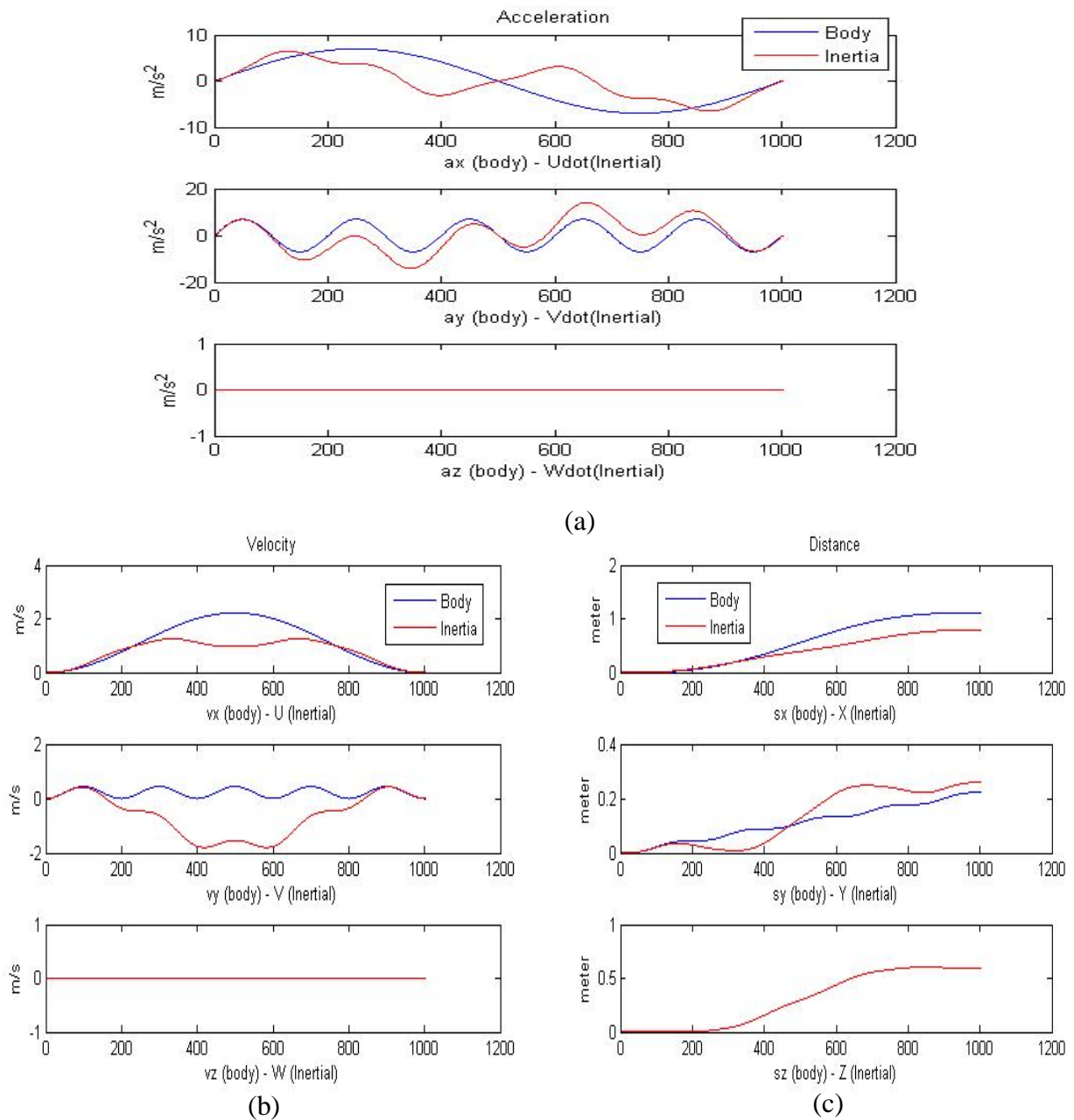
Dari grafik pada (Gambar 3-3) diketahui bahwa percepatan sumbu x dan y dari tata acuan sumbu *inertial* sudah dipengaruhi oleh adanya perubahan dari sudut *yaw*, sehingga kecepatan dan jarak juga akan mengikuti dari perubahan percepatan. Perubahan pada percepatan hanya terjadi di tengah-tengah, tetapi pada saat awal dan akhir nilai/data percepatan tetap sama, dengan percepatan 5 m/s^2 dan sudut *yaw* 100° terdapat perubahan jarak pada sumbu x = 0.3 m, sumbu y = 0.9 m, dan pada sumbu z = 0.06 m.



Gambar 3-3: Perubahan sudut ψ (*yaw*), sumbu x dan y *accelerometer* masing-masing sebesar 1 Hz dan sumbu z tidak ada perubahan. (a) percepatan, (b) kecepatan, dan (c) jarak

Benda atau kendaraan pada bidang datar bergerak hanya pada sumbu x dan y tanpa ada perubahan pada sumbu z dengan asumsi bahwa tidak ada tanjakan atau turunan pada daerah yang dilewati, sehingga pada pengujian juga membuat data *accelerometer* pada sumbu z tidak ada perubahan atau dalam keadaan diam. Sumbu x dan sumbu y masing-masing diberikan data sebesar 1 Hz seperti terlihat pada (Gambar 3-3). Sedangkan terlihat pada grafik (Gambar 3-4) di

bawah berikut ini masih sama seperti pada percobaan sebelumnya yaitu untuk ujicoba pada ruang 2D sehingga sudut *yaw*, sumbu x dan y pada *accelerometer* yang diubah tetapi diberikan pergerakan yang berbeda pada sumbu y. Sudut *yaw* tetap 100° , sumbu x = 1 Hz dan sumbu y = 5 Hz. Selisih jarak antara sumbu referensi *body* dengan *inertial* pada arah sumbu x = 0.2 m, sumbu y = 0.02 m, dan sumbu z = 0.6 m.



Gambar 3-4: Accelerometer dengan perubahan pada sumbu x sebesar 1 Hz dan sumbu y sebesar 6 Hz, (a) percepatan, (b) kecepatan, dan (c) jarak

4 KESIMPULAN

Telah dikembangkan algoritma INS untuk roket kendali LAPAN, dengan modifikasi pada metode *Quaternion* dan juga normalisasi parameter-parameter *Quaternion* dengan cara yang berbeda yang digunakan untuk roket, sehingga dari penelitian ini diperoleh beberapa kesimpulan yaitu:

- Hasil simulasi membuktikan bahwa pola dari sumbu referensi *body* dan hasil transformasi ke sumbu *inertial*

memiliki pola yang sama tetapi dengan nilai yang berbeda atau terdapat selisih nilai yang kecil antara koordinat sumbu *body* dengan koordinat sumbu *inertial*.

- Pada percepatan 5 m/s² dan semua sumbu rotasi = 0 atau tak bergerak terdapat selisih jarak sekitar 20 cm di setiap sumbu translasi.
- Pada percepatan 5 m/s² dan sudut *yaw* 100° terdapat perubahan jarak pada sumbu x = 0.3 m, sumbu y = 0.9 m, dan pada sumbu z = 0.06 m.

- Dari keseluruhan percobaan diperoleh keakuratan algoritma INS sebesar 0.2 sampai 1 meter untuk skala simulasi.

Saran dari penelitian ini adalah perlunya dilakukan ujicoba modifikasi persamaan *Quaternion* dengan menggunakan mikroprosesor berkecepatan tinggi untuk mengetahui, menguji, dan menganalisa hasil algoritma INS secara *real-time*.

DAFTAR RUJUKAN

- Flenniken, W.S., 2005. *Modeling Inertial Measurement Units and Analyzing the Effect of Their Errors In Navigation Applications*, Thesis report, Alabama, USA.
- Jia, H. 2004. *Data Fusion Methodologies for Multisensor Aircraft Navigation System*, PhD Thesis, Cranfield University.
- Kumar, V.N., 2004. *Integration of Inertial Navigation System and Global Positioning System Using Kalman Filtering*, Master Thesis, Indian Institute of Technology, Bombay, India.
- Walchko, K.J., 2002. *Low Cost Inertial Navigation: Learning to Integrate Noise and Find your Way*, Master Thesis, University of Florida, USA.
- Wiryadinata, R., T.S. Widodo, W. Widada, Sunarno, 2009. *Design of 6 DOF IMU v2.1 and GPS included INS Algorithm for Carried in Vehicle*. Proceeding of QIR UI, Depok, Indonesia.
- www.casde.iitb.ac.in/Publications/pdfdoc-2004/vikas-ddp.pdf.
- www.mil.ufl.edu/publications/thes_diss/Kevin_Walchko_thesis.pdf.