

L1-SAIF メッセージによる QZS 衛星位置の計算

08-CNS-254-1

Nov. 6, 2008

電子航法研究所

1 QZS 航法メッセージ

準天頂衛星が放送する L1-SAIF 信号は測距信号として使用できるが、これにより測位計算を実行するためには準天頂衛星の位置を計算するための情報（エフェメリス情報）が必要となる。準天頂衛星信号仕様（IS-QZSS）では、L1-SAIF 信号により放送されるメッセージのうちメッセージタイプ 58 として準天頂衛星自身のエフェメリス情報が放送されることとなっている [1]。

メッセージタイプ 58 のフォーマットを表 1 に示す。エフェメリス情報は準天頂衛星の位置と速度、加速度として記述されている。座標系は GPS で標準的に使用されている地球中心・地球固定座標系（ECEF）である。位置と速度については ECEF 座標系による値そのものが得られるが、加速度については太陽・月の重力を含む摂動項のみとされている。

2 QZS 衛星位置の計算

メッセージタイプ 58 のエフェメリス情報からエポック時刻前後の衛星位置を計算するには、速度および加速度を数値積分する必要がある [2]。

メッセージタイプ 58 と同様に ECEF 座標系における位置・速度・加速度を伝送する GLONASS については（座標系は PZ-90）、数値積分による計算方法が、GLONASS ICD の付録「A.3.1.2 Simplify of algorithm for re-calculation of ephemeris to current time」に述べられている [3][4]。まず、速度について

$$\frac{dx}{dt} = v_x \quad (1)$$

$$\frac{dy}{dt} = v_y \quad (2)$$

$$\frac{dz}{dt} = v_z \quad (3)$$

と表すことにする。これらの初期値は航法メッセージより得られる。加速度は、

$$\frac{dv_x}{dt} = -\frac{\mu_e x}{r^3} - \frac{3}{2} J_0^2 \frac{\mu_e R_e^2}{r^5} \left(1 - \frac{5 z^2}{r^2}\right) x + \omega_e^2 x + 2\omega_e v_y + \ddot{x} \quad (4)$$

$$\frac{dv_y}{dt} = -\frac{\mu_e y}{r^3} - \frac{3}{2} J_0^2 \frac{\mu_e R_e^2}{r^5} \left(1 - \frac{5 z^2}{r^2}\right) y + \omega_e^2 y - 2\omega_e v_x + \ddot{y} \quad (5)$$

$$\frac{dv_z}{dt} = -\frac{\mu_e z}{r^3} - \frac{3}{2} J_0^2 \frac{\mu_e R_e^2}{r^5} \left(1 - \frac{5 z^2}{r^2}\right) z + \ddot{z} \quad (6)$$

により求める。これらの計算式を準天頂衛星に適用する場合は、動径長 $r = \sqrt{x^2 + y^2 + z^2}$ 、地球重力定数 $\mu_e = 3.986005 \times 10^{14} \text{ m}^3/\text{s}^2$ 、地球赤道半径 $R_e = 6378137 \text{ m}$ 、地球重力ポテンシャルの二次の係数 $J_0^2 = 1082625.7 \times 10^{-9}$ 、地球自転角速度 $\omega_e = 7.2921151467 \times 10^{-5} \text{ rad/s}$ を用いる（これらの定数値は GLONASS とは若干異なる）。

表 1: 航法メッセージのフォーマット (メッセージタイプ 58)

パラメータ	ビット数	範囲	分解能	備考
$t_{0,Q}$	8	0 ~ 10740	60 s	エポック時刻
URA	4	0 ~ 15	—	測距精度指標
X_Q	26	± 42949.673 km	1.28 m	X 座標
Y_Q	26	± 42949.673 km	1.28 m	Y 座標
Z_Q	26	± 42949.673 km	1.28 m	Z 座標
$\dot{X}_Q$	24	± 4194.304 m/s	0.5mm/s	速度
$\dot{Y}_Q$	24	± 4194.304 m/s	0.5mm/s	速度
$\dot{Z}_Q$	24	± 4194.304 m/s	0.5mm/s	速度
$\ddot{X}_Q$	5	$\pm 32 \mu\text{m/s}^2$	$2 \mu\text{m/s}^2$	加速度
$\ddot{Y}_Q$	5	$\pm 32 \mu\text{m/s}^2$	$2 \mu\text{m/s}^2$	加速度
$\ddot{Z}_Q$	5	$\pm 32 \mu\text{m/s}^2$	$2 \mu\text{m/s}^2$	加速度
a_{Qf0}	22	± 1.953 ms	2^{-30} s	クロック補正
a_{Qf1}	13	± 3.725 ns	2^{-40} s/s	クロック補正
合計	212			

これらの式の第一項は向心加速度，第二項は楕円地球による重力ポテンシャルであって，また ω_e が現れる項は ECEF 座標系による取扱いのために必要なものである．摂動項 $\ddot{x}, \ddot{y}, \ddot{z}$ は一定とみなし，航法メッセージの値を用いる．

各項の大きさは，第一項および ω_e が現れる項をあわせて $\pm 1 \text{ m/s}^2$ のオーダー，また第二項が $10^{-5} \sim 10^{-4} \text{ m/s}^2$ 程度である．航法メッセージに含まれる加速度の範囲は $\pm 6.2 \times 10^{-9} \text{ km/s}^2 = \pm 6.2 \times 10^{-6} \text{ m/s}^2$ であるが，これは上式のうちの摂動項（もっとも右側の項）のみしか含まれていないことに注意が必要である．

3 プログラム例

準天頂衛星の位置を計算するためのプログラムの具体例を，次に示す（C 言語の関数として作成してある）．

次の例のプログラムにより衛星位置を計算するためには，まず最初に `set_ephemeris()` 関数を呼び出し，エフェメリス情報をセットする必要がある．その後は，`compute_satellite_position()` 関数を呼び出すことにより，任意の時刻 t における衛星位置を計算することができる．計算結果としては，ECEF 座標系による座標値がポインタ `px, py, pz` の指すアドレスに書き込まれる．

`compute_satellite_position()` 関数は，時刻ステップを `dt` として指定された時刻 t の付近まで数値積分を行い，最後は時刻 t における位置を計算するようにしてある．このような構成としたのは，測定データの時刻が変わるたびに最初から積分をし直すことを避け，また時刻 t を微妙に変えて衛星位置を計算する必要があることに配慮したためである．

```
#define MUE      3.986005e14      /* 地球重力定数 [m^3/s^2] */
#define dOMEGAe  7.2921151467e-5 /* 地球自転角速度 [rad/s] */
#define Re       6378137.0       /* 準楕円体の長半径 [m] */
#define J20      1082625.7e-9     /* 地球重力ポテンシャル係数 */
#define dt       0.01            /* 数値積分の時間ステップ [s] */

/* エフェメリス情報 */
```

```

static double    t0,px0,py0,pz0,vx0,vy0,vz0,ax0,ay0,az0;

/* 積分で使用 */
static double    t1,px1,py1,pz1,vx1,vy1,vz1,ax1,ay1,az1;
static double    t2,px2,py2,pz2,vx2,vy2,vz2,ax2,ay2,az2;

/* エフェメリス情報をセットする関数 */
void set_ephemeris(double t_epoch,double px,double py,double pz,
    double vx,double vy,double vz,double ax,double ay,double az)
{
    t0=t_epoch;
    px0=px; py0=py; pz0=pz;
    vx0=vx; vy0=vy; vz0=vz;
    ax0=ax; ay0=ay; az0=az;

    /* エフェメリス情報が更新されたことを示す */
    t1=-99999.99;
}

/* ステップ幅の積分を実行する関数 */
static void _compute_satellite_position(double dt)
{
    double    r,Jz;

    /* 積分の準備 */
    r    =sqrt(px1*px1+py1*py1+pz1*pz1);
    Jz    =-3.0/2.0*J20*MUe*Re*Re/pow(r,5.0)*(1.0-5.0*pz1*pz1/r/r);

    /* 加速度を計算する */
    ax2 =-mu_e*px1/r/r/r+Jz*px1+OMEGAe*OMEGAe*px1+2.0*OMEGAe*vy1+ax0;
    ay2 =-mu_e*py1/r/r/r+Jz*py1+OMEGAe*OMEGAe*py1-2.0*OMEGAe*vx1+ay0;
    az2 =-mu_e*pz1/r/r/r+Jz*pz1+az0;

    /* 加速度を積分して速度を得る */
    vx2 =vx1+dt*(ax1+ax2)/2.0;
    vy2 =vy1+dt*(ay1+ay2)/2.0;
    vz2 =vz1+dt*(az1+az2)/2.0;

    /* 速度を積分して位置を得る */
    px2 =px1+dt*(vx1+vx2)/2.0;
    py2 =py1+dt*(vy1+vy2)/2.0;
    pz2 =pz1+dt*(vz1+vz2)/2.0;
}

/* 衛星位置を計算する関数 */

```

```

void compute_satellite_position(double t,double *px,double *py,double *pz)
{
    /* 最初ならエフェメリス情報で初期化 */
    if (t1<0.0) {
        t1=t0;
        px1=px0;    py1=py0;    pz1=pz0;
        vx1=vx0;    vy1=vy0;    vz1=vz0;
        ax1=ax0;    ay1=ay0;    az1=az0;
    }
    /* エフェメリス情報をセットしたら「t1=0.0;」も実行しておくこと */

    /* 指定時刻まで積分する */
    if (t>t1) {
        while(t1+dt<t) {
            _compute_satellite_position(dt);
            px1=px2;    py1=py2;    pz1=pz2;
            vx1=vx2;    vy1=vy2;    vz1=vz2;
            ax1=ax2;    ay1=ay2;    az1=az2;
            t1+=dt;
        }
    } else {
        while(t1-dt>t) {
            _compute_satellite_position(-dt);
            px1=px2;    py1=py2;    pz1=pz2;
            vx1=vx2;    vy1=vy2;    vz1=vz2;
            ax1=ax2;    ay1=ay2;    az1=az2;
            t1-=dt;
        }
    }
    _compute_satellite_position(t-t1);
    *px=px2;    *py=py2;    *pz=pz2;
}

```

本プログラムの利用には特に制約は設けないが、本プログラムの利用によって生じるいかなる結果についても当所は責任を負わないこととする。

参考文献

- [1] 宇宙航空研究開発機構，準天頂衛星システムユーザインタフェース仕様書 (IS-QZSS)，Ver. 1.0, Nov. 2007.
- [2] 坂井，福島，武市，伊藤，準天頂衛星補強信号のためのエフェメリスメッセージ，電子情報通信学会論文誌，vol. J91-B, no. 3, pp. 309–316, March 2008.
- [3] *Global Navigation Satellite System GLONASS Interface Control Document*, Version 5.0, 2002.
- [4] 電子航法研究所，GLONASS 航法メッセージによる衛星位置の計算，07-CNS-36-2, June 16, 2006.

担当：CNS 領域 坂井 (END)