

反射波や回折波を含む GPS 信号を用いる移動ロボットの状態推定

九州工業大学 ○西田 健

Mobile Robot's State Estimation using Reflected and Diffracted Waves of a GPS Signal

Takeshi NISHIDA

Kyushu Institute of Technology

Abstract :

1. はじめに

Global Navigation Satellite System (GPS) は屋外移動ロボットの自己位置計測に広く利用される。一般に、信号をアンテナで直接受信できる GPS 衛星を可視衛星、それ以外を不可視衛星と呼び、4 以上の可視衛星が存在しない地上領域を衛星の陰と呼ぶ。都市部などで周囲に高い建造物が存在する状況では、GPS 信号に反射や回折が生ずることで衛星の陰が数多く発生するため、計測誤差が飛躍的に大きくなる。そこで本論文では、GPS 信号の反射波および回折波の統計的性質を用いることで、移動ロボットの状態推定精度を向上する手法を提案する。まず、反射波および回折波を含む GPS 信号は、真の位置情報と時変ガウス分布に従うノイズの和で表されると仮定する。この仮定に基づき、GPS 信号の時変バイアスを KF (Kalman filter) により推定し、さらにそれを利用したサンプリングと尤度評価を実行する PF (particle filter) を構成する。この状態推定機構により、従来は棄却されていた反射波や回折波を含む GPS 信号の効果的な利用が可能になり、衛星の陰におけるロボットの状態推定の精度向上が可能になる。

2. 移動ロボットの確率モデル

2.1 キネマティクスモデル

前輪操舵後輪駆動型の 4 輪移動ロボットを制御対象とする。このロボットはディファレンシャル式の駆動輪を有し、回転運動における後輪の滑りが発生しない。ロボットの真の状態量を、原点を O とする慣性座標系 $O-XY$ における位置と角度によって $\mathbf{x}_k \triangleq [x_k \ y_k \ \theta_k]^T = [z_k^T \ \theta_k]^T$ と定義する。ここで、 $k = 0, 1, 2, \dots$ は離散時刻、 $\mathbf{z}_k \triangleq [x_k \ y_k]^T$ はロボットの後輪車軸中央に設定されたロボットの中心の位置、 θ_k はロボットの進行方向と X 軸との角度、 l は前後車輪軸間の距離を表す。次に、動作指令入力を $\mathbf{u}_k \triangleq [v_k \ \phi_k]^T$ とする。ここで v_k は駆動輪の回転によって発生する進行方向速度であり、 ϕ_k は反時計回りを正とする前輪の操舵角を表す。このロボッ

トの離散時間状態方程式は以下で表せる [1]。

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) \\ &= \mathbf{x}_k + \begin{bmatrix} \frac{v_k}{\omega_k} \{ \sin(\theta_k + \omega_k \Delta) - \sin \theta_k \} \\ -\frac{v_k}{\omega_k} \{ \cos(\theta_k + \omega_k \Delta) - \cos \theta_k \} \\ \omega_k \Delta \end{bmatrix} \end{aligned} \quad (1)$$

ここで、 $\omega_k \triangleq (v_k/l) \tan \phi_k$ であり、 Δ はサンプリング周期である。

2.2 確率的離散状態空間モデル

現実的なロボットでは、駆動系のバックラッシュや劣化、さらに路面の傾きや滑りなどの影響により、制御入力にノイズが加わる場合がある。したがって、平均値が 0 のガウス分布に従う外乱が動作司令入力に混入するとして以下のようにモデル化する [1]。

$$\hat{v}_k \triangleq v_k + \varepsilon_v, \quad \varepsilon_v \sim \mathcal{N}(0, \alpha_1 v_k^2 + \alpha_2 \phi_k^2) \quad (2)$$

$$\hat{\phi}_k \triangleq \phi_k + \varepsilon_\phi, \quad \varepsilon_\phi \sim \mathcal{N}(0, \alpha_3 v_k^2 + \alpha_4 \phi_k^2) \quad (3)$$

ここで、 α_i ($i = 1, 2, 3, 4$) はノイズの性質を決定する正の定数である。また同様に、路面凹凸などによってロボットの方角角に外乱が加わることを考慮し、以下のように方角角の推移をモデル化する。

$$\hat{\theta}_{k+1} = \hat{\theta}_k + (\hat{\omega}_k + \varepsilon_\gamma) \Delta, \quad \varepsilon_\gamma \sim \mathcal{N}(0, \alpha_5 v_k^2 + \alpha_6 \phi_k^2)$$

ここで、 $\hat{\omega}_k \triangleq (\hat{v}_k/l) \tan \hat{\phi}_k$ 、 α_j ($j = 5, 6$) はノイズの性質を決定する正の定数である。また、記述の簡単のために $\boldsymbol{\varepsilon}_k \triangleq [\varepsilon_v \ \varepsilon_\phi \ \varepsilon_\gamma]^T$ と記述する。

以上の関係を用いると、式 (1) は確率遷移モデルとして次のように記述できる。

$$\begin{aligned} \hat{\mathbf{x}}_{k+1} &= p_f(\hat{\mathbf{x}}_{k+1} | \hat{\mathbf{x}}_k, \mathbf{u}_k) \\ &= \hat{\mathbf{x}}_k + \begin{bmatrix} \frac{\hat{v}_k}{\hat{\omega}_k} \left\{ \sin(\hat{\theta}_k + \hat{\omega}_k \Delta) - \sin \hat{\theta}_k \right\} \\ -\frac{\hat{v}_k}{\hat{\omega}_k} \left\{ \cos(\hat{\theta}_k + \hat{\omega}_k \Delta) - \cos \hat{\theta}_k \right\} \\ (\hat{\omega}_k + \varepsilon_\gamma) \Delta \end{bmatrix} \end{aligned} \quad (4)$$

ここで、 $\hat{\mathbf{x}}_k \triangleq [\hat{x}_k \ \hat{y}_k \ \hat{\theta}_k]^T$ は確率的に推移する要素を含む状態量を表す。

2.3 観測モデル

移動ロボットは、搭載した GPS アンテナによって各離散時刻における自己位置 $\mathbf{z}_k^{\text{GPS}} \triangleq [x_k^{\text{GPS}} \ y_k^{\text{GPS}}]^T$ を $\mathbf{z}_k^{\text{GPS}} = \mathbf{z}_k + \boldsymbol{\xi}_k$ として観測する。ここで

$$\boldsymbol{\xi}_k \sim \begin{cases} \mathcal{N}(r_k, \boldsymbol{\Sigma}_s) & \text{if in a shadow area} \\ \mathcal{N}(0, \boldsymbol{\Sigma}_b) & \text{otherwise} \end{cases} \quad (5)$$

であり、周囲の環境によってバイアスと分散が変化するガウス分布に従うノイズである。すなわち、衛星の陰ではバイアス r_k および分散 $\boldsymbol{\Sigma}_s = \text{diag}[\sigma_s^2, \sigma_s^2]$ のガウス分布に従い、GPS 信号が直接観測できる領域ではバイアス 0 および分散 $\boldsymbol{\Sigma}_b = \text{diag}[\sigma_b^2, \sigma_b^2]$ のガウス分布に従う。一般に、衛星の陰では GPS 信号の分散が大きくなることから $\det(\boldsymbol{\Sigma}_b) < \det(\boldsymbol{\Sigma}_s)$ である。

次に、方位角の計測は GPS 信号に含まれないため、ロボットの方位角は以下のように算出し、これを観測値とする。

$$\theta_k^{\text{GPS}} = \tan^{-1} \left(\frac{y_k^{\text{GPS}} - \hat{y}_{k-1}}{x_k^{\text{GPS}} - \hat{x}_{k-1}} \right) + \omega_{k-1} \Delta \quad (6)$$

これらの観測値をまとめて観測ベクトルとする。

$$\mathbf{h}(\hat{\mathbf{x}}_k) = \mathbf{y}_k^{\text{GPS}} \triangleq [x_k^{\text{GPS}} \ y_k^{\text{GPS}} \ \theta_k^{\text{GPS}}]^T \quad (7)$$

3. 状態量推定と制御

3.1 PF による状態推定と制御

本研究で構成するロボット走行制御系の状態変数線図を Fig.?? に示す。これは、ロボットの速度一定のもとで操舵角のみを制御することにより、指定された領域を順に通過することを目的として構成された制御系である。この制御系では、ロボットの状態推移は非線形モデルで表され、GPS 信号には時変ガウス分布に従うノイズが混入する。なお、GPS 信号に混入する観測雑音の時間変化に関する事前知識は与えられないとする。

GPS 信号のバイアスと分散が大きく変化する場合にも制御系を安定に駆動し続けるためには、衛星の陰における GPS 信号の統計的性質を状態推定器に事前知識として組み込む必要がある。そこでまず、KF を利用して出力信号から GPS 信号のバイアスを推定する。次に、衛星の陰における GPS 信号の統計的性質を組み込んだ PF により、KF の推定バイアスを利用した状態推定を行う。さらに、その PF の事後推定に基づいた制御入力を算出し、ロボットを制御する。

3.2 KF による GPS 信号のバイアス推定

式 (5) によりモデル化される出力信号は、次のようなガウス分布に従うノイズが混入する系として表される。

$$\boldsymbol{\eta}_{k+1} = \mathbf{A}\boldsymbol{\eta}_k, \quad \mathbf{z}_k^{\text{GPS}} = \mathbf{C}\boldsymbol{\eta}_k + \mathbf{w}_k \quad (8)$$

ここで

$$\boldsymbol{\eta}_k \triangleq \begin{bmatrix} \hat{x}_k \\ \hat{v}_{xk} \\ \hat{y}_k \\ \hat{v}_{yk} \end{bmatrix}, \quad \mathbf{A} \triangleq \begin{bmatrix} 1 & \Delta & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{C} \triangleq \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}$$

$$\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}), \quad \mathbf{R} \triangleq \text{diag}[\sigma_c^2 \ \sigma_c^2]$$

であり、 \hat{v}_{xk} と \hat{v}_{yk} はそれぞれ X 軸および Y 軸方向の速度、 σ_c^2 は \mathbf{w}_k の X 軸方向と Y 軸方向の分散である。この系に対し、ガウス分布に従うノイズを除く目的で KF を適用し $\boldsymbol{\eta}_k$ を推定する。ここで、より大きな分散に対応するために、前述の仮定より $\sigma_c^2 = \sigma_s^2$ とおく。さらに推定を利用して、GPS 信号のバイアスを $\hat{r}_k = \sqrt{\hat{x}_k^2 + \hat{y}_k^2}$ と推定する。

3.3 サンプルング

PF は M 個の重み付けされた粒子集合 $\{\mathbf{x}_k^{(i)}, \pi_k^{(i)}\}_{i=1}^M$ により状態推定を行う。ここで、 $\mathbf{x}_k^{(i)}$ は仮説を表す粒子の状態空間中の位置、 $\pi_k^{(i)} \geq 0$ は粒子の重みである。PF による推定の手続きは、サンプルング、粒子の尤度評価、リサンプルングの 3 ステップから成る。ここでは、サンプルングについて詳細に述べる。

本研究では、粒子の移動と分布のための提案分布を次のように設計する。

$$\tilde{\mathbf{x}}_k^{(i)} \sim p_q \left(\tilde{\mathbf{x}}_k^{(i)} | \mathbf{x}_{k-1}^{(i)} \right) = \begin{cases} p_f \left(\tilde{\mathbf{x}}_k^{(i)} | \mathbf{x}_{k-1}^{(i)} \right) & (95\%) \\ p_r \left(\tilde{\mathbf{x}}_k^{(i)} | \mathbf{x}_{k-1}^{(i)} \right) & (5\%) \end{cases}$$

このサンプルングにおいて、 $p_f(\cdot)$ は全体の 95% の粒子は前時刻 $k-1$ の粒子分布と入力、システムノイズを考慮した式 (4) に基づくサンプルングを行う。一方、全体からランダムに選択された 5% の粒子は、以下に示すような $p_r(\cdot)$ に従うサンプルングを行う。

$$p_r \left(\mathbf{x}_k^{(i)} | \mathbf{x}_{k-1}^{(i)} \right) = \mathbf{f} \left(\mathbf{x}_{k-1}^{\text{est}}, \mathbf{u}_{k-1} \right) + \mathbf{R}_\psi \boldsymbol{\rho}(\hat{r}_k) \quad (9)$$

ここで、 $\mathbf{x}_{k-1}^{\text{est}}$ は前時刻の事後推定、

$$\boldsymbol{\rho}(\hat{r}_k) \triangleq [\hat{r}_k \ 0]^T, \quad \boldsymbol{\rho}(\hat{r}_k) \sim \mathcal{N}(\hat{r}_s, \sigma_b^2) \quad (10)$$

$$\mathbf{R}_\psi \triangleq \begin{bmatrix} \cos \psi & -\sin \psi \\ \sin \psi & \cos \psi \end{bmatrix}, \quad \psi \in \text{rand}[0, 2\pi] \quad (11)$$

であり、 $p_q(\cdot)$ でサンプルングされた粒子の重みは次のように与える。

$$\pi_k^{(i)} = \frac{1}{M\sqrt{2\pi\sigma_s^2}} \exp \left\{ -\frac{1}{2} \frac{\boldsymbol{\rho}(\hat{r}_k)^2}{\sigma_s^2} \right\} \quad (12)$$

すなわち、式 (1) のロボットの運動モデルに基づいて推定された状態を基準として、 $\mathcal{N}(\hat{r}_k, \sigma_b^2)$ に従ってサンプルングされた粒子を半径 \hat{r}_k の同心円状にランダムに回転座標変換する。これは、衛星の陰におけるロボットの状態推定の精度を向上させるためのサンプルングの工夫である。これらのサンプルング手法の概要を Fig.?? に示す。

3.4 尤度評価

GPS 衛星の直接波が観測できる領域での GPS 信号に基づく尤度関数を以下のように与える.

$$p_{\text{direct}}(\mathbf{y}_k^{\text{GPS}} | \mathbf{x}_k^{(i)}) = \frac{2}{3} \cdot \frac{1}{2\pi\sigma_b^2} \left\{ \exp\left(-\frac{d_x^2}{2\sigma_b^2}\right) + \exp\left(-\frac{d_y^2}{2\sigma_b^2}\right) \right\} + \frac{1}{3} \cdot \frac{1}{2\pi\sigma_p^2} \exp\left(-\frac{d_\theta^2}{2\sigma_p^2}\right)$$

ここで, $d_x = \delta(x_k^{(i)} - x_k^{\text{GPS}})$, $d_y = \delta(y_k^{(i)} - y_k^{\text{GPS}})$, $d_\theta = \delta(\theta_k^{(i)} - \theta_k^{\text{GPS}})$ であり, $\delta(\cdot)$ はディラクのデルタ関数である. さらに, GPS 衛星の陰領域での GPS 信号に基づく尤度関数を以下のように与える.

$$p_{\text{shadow}}(\mathbf{y}_k^{\text{GPS}} | \mathbf{x}_k^{(i)}) = \frac{2}{3} \cdot \frac{1}{2\pi\sigma_s^2} \exp\left(-\frac{d_{xy}^2}{2\sigma_s^2}\right) + \frac{1}{3} \cdot \frac{1}{2\pi\sigma_p^2} \exp\left(-\frac{d_\theta^2}{2\sigma_p^2}\right)$$

ここで, $d_{xy} = |\delta(z_k^{(i)} - z_k^{\text{GPS}})| - \hat{r}_k$ である. これらのモデルの合成により, GPS 信号の観測モデルを以下のように構築する.

$$p_h(\mathbf{y}_k^{\text{GPS}} | \mathbf{x}_k^{(i)}) = \frac{1}{2} p_{\text{direct}}(\mathbf{y}_k^{\text{GPS}} | \mathbf{x}_k^{(i)}) + \frac{1}{2} p_{\text{shadow}}(\mathbf{y}_k^{\text{GPS}} | \mathbf{x}_k^{(i)})$$

また, この尤度関数を利用して粒子 $\tilde{\mathbf{x}}_k^{(i)}$ の重み $\tilde{\pi}_k^{(i)}$ を次のように更新する.

$$\tilde{\pi}_k^{(i)} \propto \pi_{k-1}^{(i)} p_h(\mathbf{y}_k^{\text{GPS}} | \mathbf{x}_k^{(i)}), \quad (\forall i) \quad (13)$$

ただし, この更新の後に $\sum_{i=1}^M \tilde{\pi}_k^{(i)} = 1$ となるようにすべての重みを正規化する.

以上の尤度評価の後に PF の事後推定が以下のように算出できる.

$$\mathbf{x}_k^{\text{est}} = \sum_{i=1}^M \pi_k^{(i)} \delta(\mathbf{x}^{(i)}) \quad (14)$$

さらに, この事後推定を用いて制御入力 $\phi_{k+1} = k_\phi(\phi_k^* \ominus \phi_k)$ と導出される [2]. ここで, $v_k = \text{const.}$ および $\phi_k^* = \tan^{-1}\{(y_i^{\text{target}} - y_k^{\text{est}}) / (x_i^{\text{target}} - x_k^{\text{est}})\}$, $\phi^*, \phi \in [0, 2\pi)$ であり, \ominus は $[0, 2\pi)$ における小さい方の角度差, k_ϕ はフィードバックゲインである.

3.5 リサンプリング

有効サンプルサイズ (ESS: effective sample size) [3] に基づきリサンプリングの実行を判断する.

$$ESS = \frac{1}{\sum_{i=1}^M (\tilde{\pi}_k^{(i)})^2} \quad (15)$$

この値は, 全粒子の重みが均等である場合に $ESS = M$ となり, 重みの偏りが最も大きい場合に $ESS = 1$ となる. 適当なしきい値 ESS_{th} を設け, ESS の値がそれを下回ればリサンプリングを実行する. すなわち, $\tilde{\pi}_k^{(i)}$ の確率で $\mathbf{x}_k^{(i)}$ を復元抽出する.

$$\mathbf{x}_k^{(i)} \sim \begin{cases} \tilde{\mathbf{x}}_k^{(1)} & \text{with prob. } \tilde{\pi}_k^{(1)} \\ \vdots & \vdots \\ \tilde{\mathbf{x}}_k^{(M)} & \text{with prob. } \tilde{\pi}_k^{(M)} \end{cases}, \quad (\forall i) \quad (16)$$

その後, 重みを $\pi_k^{(i)} := 1/M$, $(\forall i)$ として均等化する. ここで “:=” は代入を意味する. リサンプリングを行わない場合には, $\mathbf{x}_k^{(i)} := \tilde{\mathbf{x}}_k^{(i)}$, $\pi_k^{(i)} := \tilde{\pi}_k^{(i)}$ とする. 以上の処理によって新しい時刻の粒子の集合 $\{(\mathbf{x}_k^{(m)}, \pi_k^{(m)})\}_{m=1}^M$ が獲得される. この後に $k := k + 1$ としてサンプリング手順に戻る.

4 シミュレーション

ロボットはフィールド上に設定された 8 つのターゲットを順番に通過するよう制御される. ただし, ターゲットから 0.5[m] 以内にロボットが近接したと推定されたら, 目標ターゲットを次の順番のターゲットに移す. 走行速度は $v_k = 10[\text{m/s}]$ で一定とした. さらに, ロボット固有の各パラメータは次のように定めた: 車体の長さ $l = 1[\text{m}]$, 移動における速度成分の雑音パラメータ $\alpha_1 = \alpha_3 = \alpha_5 = 3.0$, 操舵角成分の雑音パラメータ $\alpha_2 = \alpha_4 = \alpha_6 = 0.3$, フィードバックゲイン $k_\phi = 0.8$, サンプリングおよび制御周期 $\Delta = 0.01[\text{s}]$.

次に, GPS 信号の雑音に関する各パラメータは次のように定めた: $\sigma_b^2 = 0.0025$, $\sigma_s^2 = 0.1$, $\sigma_p^2 = 0.01$. さらに, フィールドの $X < 0$ の領域が GPS 信号の影領域であるとし, GPS 信号に加わるノイズのバイアスを $r_k = 4[\text{m}]$ (const.) とした. PF の粒子数は $M = 1000$, ESS のしきい値は $ESS_{\text{th}} = 5.0$ とした.

これらの図より, 提案手法では衛星の陰において, キネマティクスモデルに基づく粒子分布とバイアス推定に基づく同心円状の粒子分布が同時に存在することがわかる. このような粒子分布によって, 反射や回折によって生ずる GPS 信号の誤差を考慮することが可能になっている.

次に, 次のような一般的な PF の構成によるシミュレーションを比較のために行った. すなわち, サンプリング

グはシキネマティクスモデルのみに従う.

$$\tilde{\mathbf{x}}_k^{(i)} \sim p_q \left(\tilde{\mathbf{x}}_k^{(i)} | \mathbf{x}_{k-1}^{(i)} \right) = p_f \left(\tilde{\mathbf{x}}_k^{(i)} | \mathbf{x}_{k-1}^{(i)} \right) \quad (17)$$

さらに, GPS 信号が直接観測できる場合の尤度関数のみを利用する.

$$p_h \left(\mathbf{y}_k^{\text{GPS}} | \mathbf{x}_k^{(i)} \right) = p_{\text{direct}} \left(\mathbf{y}_k^{\text{GPS}} | \mathbf{x}_k^{(i)} \right) \quad (18)$$

提案手法と従来手法によるロボットの走行軌跡を, それぞれ Fig.??(a) と (b) に示す. これらより, 提案手法に基づく走行は, 時間経過の後にも状態推定の精度が高く走行軌跡にも摂動が少ない. 一方, 従来手法に基づく走行には, 時間経過に伴う誤差の累積や走行の不安定化が見られる.

5. おわりに

GPS 計測に KF と PF を連動させて適用し, 移動ロボットの自己位置の推定精度を向上させる手法を提案した. まず, 反射波および回折波を含む GPS 信号は, 真の位置情報と時変ガウス分布に従うノイズの和で表されると仮定して系の定式化を行った. 次に, その仮定に基づき, GPS 信号の時変バイアスを KF により推定し, それを利用したサンプリングと尤度評価を実行する PF を構成した. 最後に, これを組み込んだ自律走行ロボットの走行制御系が有効に動作することをシミュレーションより示した.

参考文献

- [1] S. Thrun, W. Burgard and D. Fox (2005) Probabilistic Robotics. MIT Press.
- [2] P. Corke (2011) Robotics, Vision and Control Fundamental Algorithms in MATLAB, Springer-Verlag.
- [3] J. S. Liu and R. Chen (1998) Sequential Monte Carlo methods for dynamic systems, Journal of the American Statistical Association, Vol. 93, No. 443, pp. 1032–1044.