

拡張カルマン粒子フィルターに基づく相手ロボットの位置推定

Predicting positions of opponent robots based on Extended Kalman Particle Filter

上野滋弘, 村上和人, 成瀬正

Shigehiro Ueno, Kazuhito Murakami and Tadashi Naruse

愛知県立大学大学院

Graduate school of Information Science and Technology, Aichi Prefectural University

概要

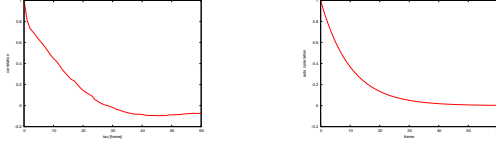
This paper describes a method that predicts the future positions of the opponent robots precisely using the Extended Kalman Particle Filter. In the RoboCup Small Size League, the positions of the teammate and opponent robots and the ball dictate the team's strategy. The positions are given by the global vision system, but they have noises. Moreover, since the actions of the teammate robots are computed based on the past, current and future positions of the robots on the field, it is necessary to get future positions of the opponent robots as precise as possible. We don't know the opponent strategy so that we have to predict the future positions of opponent robots from the past position data. So, we propose a following method to predict the positions of opponent robots. First, we assume that the acceleration of opponent robot is a Gauss-Markov process. When we write a state model as $\dot{\mathbf{x}} = \mathbf{F}\mathbf{x} + \mathbf{G}\mathbf{u}$ and $\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{v}$, the matrices \mathbf{F} and \mathbf{G} have a common parameter β which should be estimated. We estimate it based on the Extended Kalman Particle Filter. Experimental result shows that the mean square error (MSE) of the predicted positions reduced to 55% or less compared with the MSE of the predicted positions which is used in our team so far.

1 はじめに

RoboCup 小型リーグでは, 複数台のロボット間でパスやシュートなどの連携動作をベースにした試合が展開されている [1][2]. このような高度な連携動作を行うためには, 味方ロボットやボールの位置だけでなく, 相手ロボットの高精度な位置予測を行うことが求められている. しかしながら, グローバルビジョンシステムより得られる位置情報には, カメラの空間分解能の制限により, ノイズが含まれている. また, 近年の小型リーグでのフィールドサイズの拡大により, 観測位置精度の低下が懸念されている. 相手チームの戦略は分からないため, 現在までのノイズを含んだ観測値のみで, 予測位置を計算しなければならない.

従来手法では, 相手ロボットの運動が等速直線運動に入力が加わるモデルを用い, Kalman Filter(KF)[3]を用いて, 状態推定および予測を行っていた. ここで, 入力は, 環境からロボットに与えられる雑音成分だけでなく, 相手チームの戦略によって決定される加速度も含んでいる. これを, 平均 0, 分散 σ_a^2 のガウス白色過程と仮定している. しかしながら, 近年のロボット性能の飛躍的な向上により, 加速度が 3m/s^2 を越えることもあるため, 従来手法では分散 σ_a^2 が非常に大きな値となる. これは予測精度の低下に繋がる. また, 実際にシミュレータを用いて加速度を計測し, その自己相関をとると図 1(a) のような結果が得られる. もし, 入力が白色過程であれば,

その自己相関関数はディラックの δ 関数となるので、離散時間での状態方程式を示す．
白色過程と仮定するのは無理がある．



(a) ロボットの加速度 (b) Gauss-Markov 過程

図 1: 自己相関関数

一方，Gauss-Markov 過程の自己相関関数は，図 1(b) となる．このことから，入力を Gauss-Markov 過程と仮定してモデル化することが適切であると考えられる．このモデルにおいては，入力の相関時間を表すパラメータが存在する．相関時間が長いほど等加速度運動に近付き，短いほど白色性の入力となる．このパラメータの値は，相手チームの戦略およびロボットの性能に強く依存するため，状態と同様に推定を行う必要がある．このため，本稿では，Extended Kalman Filter(EKF)[4] および Particle Filter(PF)[5] を組み合わせた Extended Kalman Particle Filter(EKPF) に基づき推定する手法を提案する．また，シミュレーションにて生成した真値と観測値から，本手法と従来手法での予測誤差を比較し，本手法の有効性を示す．

2 状態空間モデル

位置の推定および予測を行うために，ロボットの運動を状態空間モデルを用いて表現する．ただし， x 方向と y 方向の動きは独立であることを仮定し，表記の簡略化のために，一方向についてのみ記述する．

2.1 既存手法の状態空間モデル

位置 (loc) および速度 (vel) を要素とする二次元ベクトルを状態として定義し，等速直線運動に白色性の入力加わるものとしてモデル化を行う．以下に

$$\begin{bmatrix} \text{loc}_{k+1} \\ \text{vel}_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \text{loc}_k \\ \text{vel}_k \end{bmatrix} + \mathbf{w}_k \quad (1)$$

$$z_k = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \text{loc}_k \\ \text{vel}_k \end{bmatrix} + v_k \quad (2)$$

ここで， z_k は観測位置の座標である． v_k は観測ノイズであり，分散 σ_v^2 の離散白色ガウスノイズである． \mathbf{w}_k は加速度による変化を表す項であり，以下の性質を満たす．

$$E[\mathbf{w}_k \mathbf{w}_j^T] = \begin{cases} \mathbf{Q}_k & (\text{for } k = j) \\ 0 & (\text{for } k \neq j) \end{cases} \quad (3)$$

2.2 提案手法の状態空間モデル

Gauss-Markov 過程の自己相関関数は以下のように定義される．

$$R_a(\tau) = E[\{a(t+\tau) - a(t)\}^2] = \sigma_a^2 e^{-\beta|\tau|} \quad (4)$$

ここで， $a(t)$ のスペクトル関数を $A(s)$ とし，以下の式を満たす伝達関数 $G(s)$ を決定する．

$$A(s) = G(s)U(s) \quad (5)$$

ここで， $U(s)$ が単位白色過程のスペクトルとすると， $U(s) = 1$ である．式 (4) をフーリエ変換すると，ウィナー・ヒンチンの定理より，

$$\begin{aligned} S_R &= \frac{2\beta\sigma_a^2}{(s+\beta)(s-\beta)} \\ &= \frac{\sqrt{2\beta\sigma_a^2}}{s+\beta} \frac{\sqrt{2\beta\sigma_a^2}}{s-\beta} \\ &= S^- \cdot S^+ \end{aligned} \quad (6)$$

となる．解の安定性のため，極が左半面にある S^- を伝達関数 $G(s)$ とすると，加速度の満たすべき微分方程式は式 (7) となる．

$$\dot{a} = -\beta a + \sqrt{2\beta\sigma_a^2}u \quad (7)$$

位置・速度・加速度を要素とする三次元ベクトル \mathbf{x} を状態とすると，式 (7) から以下の状態方程式が導

かれる．

$$\dot{\mathbf{x}} = \mathbf{F}\mathbf{x} + \mathbf{B}u = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -\beta \end{bmatrix} \begin{bmatrix} \text{loc} \\ \text{vel} \\ \text{acc} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \sqrt{2\beta\sigma_a^2} \end{bmatrix} u \quad (8)$$

これを離散系へ変換すると，

$$\mathbf{x}_{k+1} = \phi\mathbf{x}_k + \mathbf{w}_k = \begin{bmatrix} 1 & \Delta t & \frac{1}{\beta^2}(e^{-\beta\Delta t} + \beta\Delta t - 1) \\ 0 & 1 & (1 - e^{-\beta\Delta t}) \\ 0 & 0 & e^{-\beta\Delta t} \end{bmatrix} \begin{bmatrix} \text{loc}_k \\ \text{vel}_k \\ \text{acc}_k \end{bmatrix} + \mathbf{w}_k \quad (9)$$

となる．ここで， \mathbf{w}_k は以下の性質を満たす．

$$E[\mathbf{w}_k] = \mathbf{0} \quad (10)$$

$$E[\mathbf{w}_k \mathbf{w}_j^T] = \begin{cases} \mathbf{Q}(\beta) & (\text{for } k = j) \\ 0 & (\text{for } k \neq j) \end{cases} \quad (11)$$

観測方程式は，ノイズを含んだ位置情報のみが得られるため，以下のように定める．

$$z_k = \mathbf{H}\mathbf{x}_k + v_k = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \text{loc}_k \\ \text{vel}_k \\ \text{acc}_k \end{bmatrix} + v_k \quad (12)$$

v_k は白色性の観測ノイズであり，平均0，分散 σ_v^2 とする．

2.3 パラメータの推定モデル

式(4)において， β は相関時間を表している．このパラメータ β を推定するため，定常性を仮定し，以下の状態方程式および観測方程式を定めた．

$$\beta_{k+1} = \beta_k \quad (13)$$

$$z_k = h(\hat{\mathbf{x}}_{k-1}, \beta_k) + v_{\beta,k} = \mathbf{H}\phi(\beta_k)\hat{\mathbf{x}}_{k-1} + v_{\beta,k} \quad (14)$$

ここで z_k は式(12)と同じ観測位置である． $\hat{\mathbf{x}}_{k-1}$ は一時刻前の状態の事後推定値， $v_{\beta,k}$ は平均0，分散 $\sigma_{\beta,v}^2$ のガウス白色雑音である．

3 推定および予測手法

3.1 提案モデルに対する状態推定手法

エルゴード性が成立すると仮定し，Kalman フィルタを用いて状態推定を行う． β_k は未知であるため，事前推定値 $\hat{\beta}_k^-$ に置き換え，以下の式によって逐次的に推定する．

$$\hat{\mathbf{x}}_k^- = \phi(\hat{\beta}_k^-)\hat{\mathbf{x}}_{k-1} \quad (15)$$

$$\mathbf{P}_k^- = \phi(\hat{\beta}_k^-)\mathbf{P}_{k-1}(\phi(\hat{\beta}_k^-))^T + \mathbf{Q}(\hat{\beta}_k^-) \quad (16)$$

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}^T [\mathbf{H}\mathbf{P}_k^- \mathbf{H}^T + \sigma_v^2]^{-1} \quad (17)$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k(z_k - \mathbf{H}\hat{\mathbf{x}}_k^-) \quad (18)$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H})\mathbf{P}_k^- \quad (19)$$

また， M フレーム先の状態を予測する場合には，次式を用いて計算する．

$$\hat{\mathbf{x}}_{k+M} = \left(\phi(\hat{\beta}_k^-)\right)^M \hat{\mathbf{x}}_k \quad (20)$$

3.2 提案モデルに対するパラメータ推定手法

式(14)の関数 h は式(9)より， β に関して非線形な関数となっていることが分かる．Extended Kalman Filter(EKF)[4] では，非線形な関数を線形近似し，KFを用いて状態推定を行う．しかしながら，このモデルにおいては β を直接観測することができないため，位置の観測値 z_k を用いて更新を行わなければならない．よって，EKFの一次近似の考えを基に，パラメータ β を推定する手法を示す．

$h(\cdot)$ を $\hat{\beta}_k^-$ のまわりでテイラー展開すると，

$$h(\hat{\mathbf{x}}_{k-1}, \beta_k) = h(\hat{\mathbf{x}}_{k-1}, \hat{\beta}_k^-) + \left. \frac{\partial h(\hat{\mathbf{x}}_{k-1}, \beta)}{\partial \beta} \right|_{\beta=\hat{\beta}_k^-} (\beta_k - \hat{\beta}_k^-) + \dots \approx \mathbf{H}\hat{\mathbf{x}}_{k-1} + \mathbf{H} \left(\frac{\partial \hat{\mathbf{x}}_{k-1}}{\partial \beta} + \frac{\partial \phi(\beta)}{\partial \beta} \hat{\mathbf{x}}_{k-1} \right) \Big|_{\beta=\hat{\beta}_k^-} (\beta_k - \hat{\beta}_k^-) \quad (21)$$

となる．以降，上記第二項の係数を下記のように略記する．

$$H_k^\beta = \mathbf{H} \left(\frac{\partial \hat{\mathbf{x}}_{k-1}}{\partial \beta} + \frac{\partial \phi(\beta)}{\partial \beta} \hat{\mathbf{x}}_{k-1} \right) \Big|_{\beta=\hat{\beta}_k^-} \quad (22)$$

このモデルに Kalman フィルタを適用することを考える．事前推定値から事後推定値への更新を次式で行うものとする．

$$\hat{\beta}_k = \hat{\beta}_k^- + K_{\beta_k}(z_k - \mathbf{H}\hat{\mathbf{x}}_k^-) \quad (23)$$

そうすると，式 (21),(22),(23) より，誤差共分散行列 \mathbf{P}_{β_k} は以下のように変形できる．

$$\begin{aligned} \mathbf{P}_{\beta_k} &= E \left[(\beta_k - \hat{\beta}_k)^2 \right] \\ &= E \left[\left\{ (\beta_k - \hat{\beta}_k) - K_{\beta_k}(z_k - \mathbf{H}\hat{\mathbf{x}}_k^-) \right\}^2 \right] \\ &= (1 - K_{\beta_k} H_k^\beta)^2 P_{\beta_k}^- + (K_{\beta_k})^2 \sigma_{v,\beta}^2 \end{aligned} \quad (24)$$

これを K_{β_k} で微分したものが極小となるようにするため， $\frac{\partial P_{\beta_k}}{\partial K_{\beta_k}} = 0$ を解くと，次式を得る．

$$K_{\beta_k} = P_{\beta_k}^- H_k^\beta \left[(H_k^\beta)^2 P_{\beta_k}^- + \sigma_{v,\beta}^2 \right]^{-1} \quad (25)$$

これらをまとめて，式 (15)～式 (19) に対応する式を示すと，以下のようになる．

$$\hat{\beta}_k^- = \hat{\beta}_{k-1} \quad (26)$$

$$P_{\beta_k}^- = P_{\beta_{k-1}} + \sigma_{w,\beta}^2 \quad (27)$$

$$K_{\beta_k} = P_{\beta_k}^- H_k^\beta \left[(H_k^\beta)^2 P_{\beta_k}^- + \sigma_{v,\beta}^2 \right]^{-1} \quad (28)$$

$$\hat{\beta}_k = \hat{\beta}_k^- + K_{\beta_k}(z_k - \mathbf{H}\hat{\mathbf{x}}_k^-) \quad (29)$$

$$P_{\beta_k} = (1 - K_{\beta_k} H_k^\beta) P_{\beta_k}^- \quad (30)$$

この式を用いて逐次的に β を推定する．また， H_k^β に関して以下を用いて計算することができる．

$$\frac{\partial \hat{\mathbf{x}}_{k+1}^-}{\partial \beta} = \phi(\hat{\beta}_k) \frac{\partial \hat{\mathbf{x}}_k}{\partial \beta} + \frac{\partial \phi(\beta)}{\partial \beta} \Big|_{\beta=\hat{\beta}_k} \hat{\mathbf{x}}_k \quad (31)$$

$$\begin{aligned} \frac{\partial \hat{\mathbf{x}}_k}{\partial \beta} &= (\mathbf{I} - \mathbf{K}_k \mathbf{H}) \frac{\partial \hat{\mathbf{x}}_k^-}{\partial \beta} \\ &\quad + \frac{\partial \mathbf{K}_k}{\partial \beta} (z_k - \mathbf{H}\hat{\mathbf{x}}_k^-) \end{aligned} \quad (32)$$

$$\frac{\partial \mathbf{K}_k}{\partial \beta} = \frac{\mathbf{I} - \mathbf{K}_k \mathbf{H}}{\mathbf{H} \mathbf{P}_k^- \mathbf{H}^T + \sigma_u^2} \frac{\partial \mathbf{P}_k^-}{\partial \beta} \mathbf{H}^T \quad (33)$$

$$\begin{aligned} \frac{\partial \mathbf{P}_k^-}{\partial \beta} &= \frac{\partial \phi(\beta)}{\partial \beta} \Big|_{\beta=\hat{\beta}_k^-} \mathbf{P}_{k-1} \phi^T(\hat{\beta}_k^-) \\ &\quad + \phi(\hat{\beta}_k^-) \frac{\partial \mathbf{P}_{k-1}}{\partial \beta} \phi^T(\hat{\beta}_k^-) \\ &\quad + \phi(\hat{\beta}_k^-) \mathbf{P}_{k-1} \frac{\partial \phi^T(\beta)}{\partial \beta} \Big|_{\beta=\hat{\beta}_k^-} \end{aligned} \quad (34)$$

$$\begin{aligned} \frac{\partial \mathbf{P}_{k-1}}{\partial \beta} &= -\frac{\partial \mathbf{K}_{k-1}}{\partial \beta} \mathbf{H} \mathbf{P}_{k-1}^- \\ &\quad + (\mathbf{I} - \mathbf{K}_{k-1} \mathbf{H}) \frac{\partial \mathbf{P}_{k-1}^-}{\partial \beta} \end{aligned} \quad (35)$$

3.3 Particle Filter の適用と位置予測

パラメータの推定に対して，一次近似によって推定を行う手法を示してきた．しかしながら，与える初期値によっては， $\beta_k - \hat{\beta}_k$ が大きな値となってしまう，2 次以上の項が無視できなくなることが考えられる．そのため，Particle Filter(PF) を用いて，2 次以上の項の影響を削減する．

以下に，予測アルゴリズムを示す．ただし，状態 \mathbf{x} およびパラメータ β の組を 1 つのパーティクルとする．

1. 式 (15) を用いて状態を更新する．
2. N 個のパーティクルの $\hat{\beta}_k^{(i)}$ ($i = 1, \dots, N$) に対し， $N(0, P_{\beta,k}^{(i)})$ に従う乱数を加える (Sampling) ．
3. それぞれのパーティクルに対し，式 (16)-(19) および (26)-(35) を計算する
4. それぞれのパーティクルの $\hat{\mathbf{x}}_k^{(i)}$ に対し次式を用いて尤度を計算する．

$$w_k^{(i)} = p(z_k | \hat{\mathbf{x}}_k^{(i)}) = \frac{1}{\sqrt{2\pi\sigma_v^2}} e^{-\frac{(z_k - \mathbf{H}\hat{\mathbf{x}}_k^{(i)})^2}{2\sigma_v^2}} \quad (36)$$

5. w を正規化する．

$$\tilde{w}_k^{(i)} = \frac{w_k^{(i)}}{\sum_{j=1}^N w_k^{(j)}} \quad (37)$$

6. 5 で求めた重みを用いて，事後分布を求める．提案手法では sampling importance resampling(SIR)[6] を適用する．よって，確率 $\{\tilde{w}_k^{(i)} | i = 1, \dots, N\}$ でパーティクル $\{(\hat{\mathbf{x}}_k^{(i)}, \hat{\beta}_k^{(i)}) | i = 1, \dots, N\}$ をサンプリングし，新たな N 個のパーティクルの集合を生成する．
7. 状態の平均値を推定値 $\hat{\mathbf{x}}_k^*$ とする．同様に $\hat{\beta}_k^{(i)}$ の平均値を求め，その値を $\hat{\beta}_k^*$ とする．
8. 予測するフレーム数を M とすると，以下の式によって，予測値を求める．

$$\hat{\mathbf{x}}_{k+M}^* = (\phi(\hat{\beta}_k^*))^M \hat{\mathbf{x}}_k^* \quad (38)$$

9. 1～8 を繰り返す．

4 評価実験

4.1 実験の方法と結果

実際の試合では、ノイズを含む観測値のみが得られるため、真値を求めることができない。そのため、シミュレータ上で5分間の試合を行わせ、そのときのロボットの位置の系列を真値とした。また、その真値データにガウス白色雑音を加えた観測値を生成した。シミュレーションは、愛知県立大学 RoboDragons チームで使用されている戦略を用い、60fps で処理を行う。ロボットの内部モデルは一次遅れ系とした。

既存手法および提案手法に観測値を与え、予測を行わせた。戦略に関する情報は与えないこととした。提案手法におけるパーティクル数は $N = 100$ とした。平均二乗誤差 (MSE) および誤差分布を求め、提案手法の有効性を確認した。

3 パターンの試合データに対し、推定および5~20フレーム先の予測を行った。既存手法の MSE を 1.0 とした場合の提案手法の MSE を表 1 に示す。また、Pattern1 の5フレーム先の予測に対する従来手法および提案手法の誤差分布を図 2 および図 3 に示す。

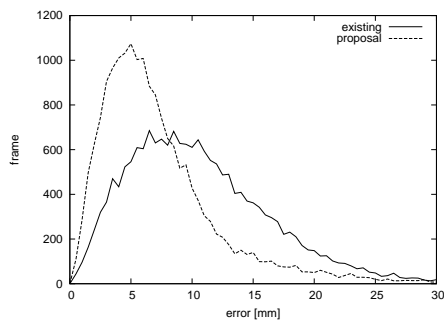


図 2: 5 フレーム先予測に対する誤差分布

4.2 考察

実験結果から、全てのパターンにおいて、提案手法がより誤差の少ない予測を行っていることが分かる。特に、RoboDragons チームのシステムで必要と

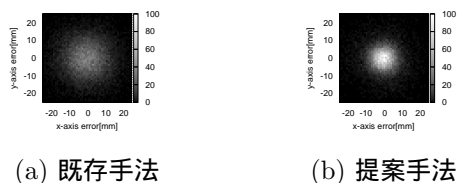


図 3: 5 フレーム先予測に対する誤差分布 (真値からの距離)

されている5フレーム先の予測データにおいては、MSE が 55% 以下になっており、その有効性を認めることができる。

また、図 3 より、誤差が同心円上に分布しているため、 x 方向と y 方向それぞれ独立な正規分布に従うことが分かる。このことは、それぞれの方向に対して独立に処理を行うことが妥当であることを示していると考えられる。

パラメータ β の推定にパーティクルフィルタを用いた場合と、用いない場合の比較を行った。図 4 に Pattern1 における β の推定結果を示す。

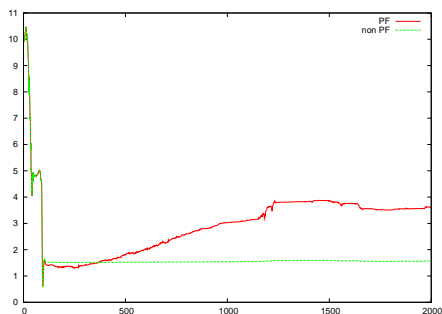


図 4: PF の使用による β 推定の変化

Pattern1 の真値 (18000frame) から求めた時間平均の自己相関は、 $\beta = 5.0$ の Gauss-Markov 過程で近似できる。従って、パーティクルフィルタを用いた場合は、用いない場合と比較してより誤差の少ない推定を行っていることが分かる。

処理時間は、パーティクルフィルタを用いない場合は 0.2ms/frame であり、用いた場合は、 $N = 100$ で 19.5ms/frame であった。RoboDragons のシステム

表 1: 提案手法の MSE(既存手法=1.0)

| | esitimate | 5 Frame | 10 Frame | 15 Frame | 20 Frame |
|----------|-----------|---------|----------|----------|----------|
| Pattern1 | 0.732 | 0.546 | 0.652 | 0.744 | 0.778 |
| Pattern2 | 0.731 | 0.518 | 0.604 | 0.702 | 0.732 |
| Pattern3 | 0.715 | 0.514 | 0.591 | 0.680 | 0.721 |

でこの計算に許される処理時間は 5ms である。パーティクルフィルタの計算はパーティクルごとに並列に計算できるので、並列処理により、達成できると考えられる。

パーティクル数と予測精度に関しては、 $N = 50$ から $N = 100$ において最適な予測が行われていた。しかしながら、サンプリングおよびリサンプリングには乱数を用いているため、最適なパーティクル数を決定することは困難である。また、図 4 から、ある程度の時間経過後には変動が少なくなっていることがわかる。このため、一定の時間経過後はパーティクル数を減らす、またはパーティクルフィルタの使用しないといった手法が求められる。

5 おわりに

本論文では、RoboCup 小型リーグにおける相手ロボットの位置の予測手法を提案した。提案手法では、Gauss-Markov モデルで加速度をモデル化し、位置、速度、加速度からなる状態および Gauss-Markov モデルに含まれるパラメータを逐次的に推定した。評価実験の結果、等速直線運動モデルを使用した既存手法と比較し、予測精度を向上できた。

今後の課題として、相手ロボットの非定常性への対応が挙げられる。これは相手ロボットがポジションチェンジなどを行った場合に、誤差が増大するという問題があるためである。提案モデルでは Gauss-Markov モデルのパラメータ β は定常性を仮定してモデル化しているため、この問題が発生すると考えられる。

第二の課題として、実際の試合における観測系のモデル構築である。現在、観測雑音はガウス白色過

程に従うとしてモデル化を行ったが、これは静止したロボットに対する観測位置誤差の分布より求めたものである。よって、動いているロボットに対する雑音の分布を求める必要がある。

参考文献

- [1] K. Murakami, et al. "Cooperative Soccer Play by Real Small-Size Robot", In RoboCup 2003: Robot Soccer World Cup VII, LNAI3020, pp. 410 - 421, Springer-Verlag, 2003
- [2] R. Nakanishi, et al. "Cooperative 3-robots passing and shooting in the RoboCup Small Size League", RoboCup international symposium 2006, CDROM, 2006
- [3] R.E.Kalman "A new approach to linear filtering and prediction problems", Transactions of the ASME, Ser.D, Journal of Basic Engineering, Vol.82, pp.35-45, 1960
- [4] A.H. Jazwinski, "Stochastic Processes and Filtering Theory", Academic Press, 1970
- [5] J.S. Liu and R. Chen, "Sequential Monte Carlo methods for dynamic systems", Journal of the American Statistical Association, Vol.93, pp.1032-1044, 1998
- [6] D.B. Rubin, "Using the SIR algorithm to simulate posterior distributions", In J.M. Bernardo, et al. Eds., Bayesian Statistics 3, pp.395-402, Oxford University Press, 1988