

시간 지연차를 이용한 표적의 위치 추적

윤동욱*, 엄석원, 윤동헌, 고한석

고려대학교 전자공학과 지능 정보 및 신호 처리 연구실

A Position Estimation Using Time Delay Difference

Dong-uk Yoon*, Suk-won Yeom, Dong-hun Yoon, Hanseok Ko

II&SP lab, Dept. of Electronics Engineering Korea University
(duyoon, swyeom, koppie, hsko)@ispl.korea.ac.kr

요약

본 논문에서는 기존의 방법에 의해 추정된 시간 지연의 차를 이용하여 표적의 위치를 추적하는 방법을 제시한다. 표적 위치를 추정하는 알고리즘의 선단에서 시간 지연의 차를 보정하는 과정을 수행하고, 후반에서 칼만 필터로 smoothing을 하는 방법을 제시한다. 이 방법을 사용하여 최소 2N-2개의 시간 지연 차를 이용하여 표적의 위치를 보다 정확히 추적할 수 있다.

1. 서론

비자의 적 잠수함을 발견하고 위치를 추적하는 시스템의 구현은 해상무기체계의 기반을 이루는 요소이다.

표적의 위치를 계산하기 위하여, 한 쌍의 센서에 도달하는 공동된 표적신호의 시간 지연 차를 추정할 수 있다. 이러한 시간 지연은 신호의 속력이 일정하다는 가정 아래 각 센서와 잠수함과의 거리에 비례하며 여러 쌍의 센서로부터 구한 시간지연 차를 근거로 위치의 계산이 가능하다.

시간지연의 차를 이용하여 표적의 위치를 추적하는 고전적인 방법은 같은 시간지연의 차를 가지는 포물선들의 교점을 구하는 것이다. 그러나 비선형적인 방정식을 풀어야 하기 때문에 계산에 어려움이 있다.

R. O. Schmidt[1]은 3개의 센서를 지나는 타원의 초점에 표적이 있다는 사실을 이용하여 표적 추적을 하는 방법을 제시하였으나 이 역시 두 개의 초점 중 하나를 선택하는 방법에서 계산이 복잡해진다. 이 방법을 응용하여 3개의 센서에 의해서 결정되는 line of position(L.O.P)의 교점을 구하는 방법은 선형적인 방정식을 풀어 표적의 위치를 구할 수 있다.

B. Frienlander[2]: N개의 센서를 이용하여 위치를 추적하는 방법과 그에 대한 error covariance를 유도하

였다. 이 논문의 결과를 보면, 위치 추정의 오차가 관측 범위 안에 있으면서도 센서의 위치에 따라 크게 영향을 받을 수 있다.

본 논문에서는 기본적으로 [2]의 방법을 따르면서 센서의 위치에 의해 생기는 오차를 줄이기 위해서 칼만 필터를 이용하여 결과를 smoothing을 하는 방법을 제시한다. 또한 시간지연 차를 선단에서 미리 보정함으로써 계산량을 센서의 수 N에 비례하도록 유지하며 보다 정확한 결과를 얻을 수 있는 방법을 제시한다.

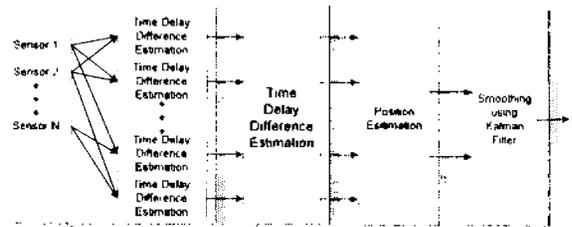


그림 1 위치 추적 알고리즘

2. 시간지연차 모델링

표적에서 방사된 신호는 시간지연을 가지고 각 센서에서 감음과 함께 관측된다. 센서들은 서로 다른 위치에 분산되어 있기 때문에, 각 센서에서 관측된 신호의 시간지연은 서로 다르다. 그러나 각 센서에서 관측된 신호 자료만 가지고는 시간지연의 차를 직접 구할 수 없고, 기존 센서로부터의 시간지연의 차만을 얻을 수 있다. 관측된 신호는 식(1)과 같이 모델링이 되며, $d_{12}(t) = d_1(t) - d_2(t)$ 만 추정이 가능하다[5].

$$\begin{aligned} x_1(t) &= s(t + d_1(t)) + n_1(t) \\ x_2(t) &= s(t + d_2(t)) + n_2(t) \end{aligned}$$

$$\begin{aligned} s(t) &: \text{표적의 신호} \\ n_i(t) &: \text{백색 가우시안 잡음} \\ d_i &: i\text{-번째 센서에서의 시간지연} \end{aligned} \quad (1)$$

시간지연의 차는 각 블록단위로 추정되며 어느 정도의 오차를 가진다고 가정을 한다. 한 쌍의 센서 사이의 시간지연의 차를 구하는 알고리즘은 각각이 독립적이므로 이때 발생하는 오차 역시 서로 독립적이다. 즉, k번째 블록에서 추정된 센서 i와 j사이의 시간지연의 차를 다음과 같이 모델링 할 수 있다.

$$\begin{aligned} \widehat{d}_{ij}(k) &= d_{ij}(k) + e_{ij}(k) \\ d_{ij}(k) &: \text{실제 시간지연차} \\ e_{ij}(k) &: \text{추정오차값} \end{aligned} \quad (2)$$

배열에서의 음파의 속도를 알고 있으면, 각 센서 사이의 시간 지연의 차에 배열에서의 음파의 속도(v_s)를 곱하여 각 센서 사이의 거리의 차를 얻을 수 있다. 즉, 표적과 센서 사이의 거리의 차는 다음과 같이 쓸 수 있다.

$$\widehat{r}_{ij} = v_s \widehat{d}_{ij}(k) + v_s e_{ij}(k) \quad (3)$$

3. 위치 추적 알고리즘

3.1 시간지연차의 보정

4 개의 센서 사이의 그림 2과 같은 방법으로 6 개의 시간지연의 차를 벡터 형태로 정리를 하면, 다음과 같다.

$$\begin{bmatrix} d_{12} \\ d_{13} \\ d_{14} \\ d_{23} \\ d_{34} \\ d_{42} \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{bmatrix} \quad (4)$$

이들 N개의 센서 사이의 2N-2개의 시간지연의 차를 이용한 식으로 일반화시키면 다음과 같다.

$$\begin{bmatrix} d_{12} \\ \vdots \\ d_{1N} \\ d_{23} \\ \vdots \\ d_{N-1N} \\ d_{N2} \end{bmatrix} = \begin{bmatrix} -\mathbf{1}_{N-1} & \mathbf{I}_{N-1} \\ \mathbf{0}_{N-1} & -\mathbf{I}_{N-1} + \mathbf{I}_N^{(1)} \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_N \end{bmatrix} \quad (5)$$

$\mathbf{1}_N$: $N \times 1$ 의 1벡터

$\mathbf{0}_N$: $N \times 1$ 의 0벡터

\mathbf{I}_N : $N \times N$ 의 단위행렬

$\mathbf{I}_N^{(1)}$: \mathbf{I}_N 를 오른쪽으로 한 칸 회전시킨 행렬

위의 식을 이용하여 다음과 같이 시간지연의 차를 보정할 수 있다.

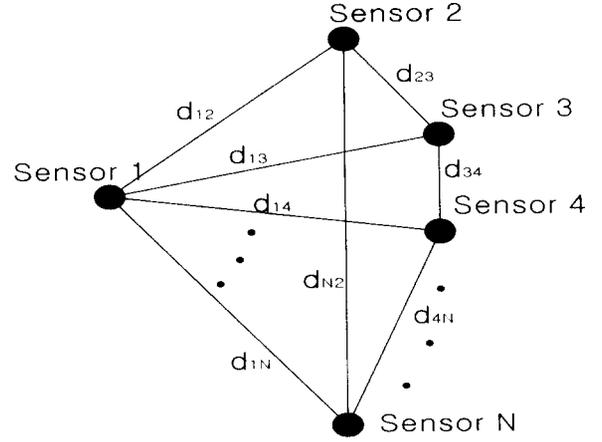


그림 2 N개 센서사이의 2N-2개의 시간지연차

$$\begin{bmatrix} \widehat{d}_{12} \\ \vdots \\ \widehat{d}_{1N-1} \\ \widehat{d}_{23} \\ \vdots \\ \widehat{d}_{N-1N} \\ \widehat{d}_{N2} \end{bmatrix} = \mathbf{D}(\mathbf{D}^T \mathbf{D})^{-1} \mathbf{D}^T \begin{bmatrix} \widehat{d}_{12} \\ \vdots \\ \widehat{d}_{1N-1} \\ \widehat{d}_{23} \\ \vdots \\ \widehat{d}_{N-1N} \\ \widehat{d}_{N2} \end{bmatrix} \quad (6)$$

$$\mathbf{D} = \begin{bmatrix} -\mathbf{1}_{N-1} & \mathbf{I}_{N-1} \\ \mathbf{0}_{N-1} & -\mathbf{I}_{N-1} + \mathbf{I}_N^{(1)} \end{bmatrix}$$

3.2 위치 추정 알고리즘 [2]

표적과 i번째 센서와의 거리를 R_{it} , 표적의 위치 벡터를 $\mathbf{x}_t = [x_t \ y_t]^T$, i 번째 센서의 위치 벡터를 $\mathbf{x}_i = [x_i \ y_i]^T$ 이라 정의하고 다음의 값을 정의하자.

$$\begin{aligned} r_{ij} &= R_{it} - R_{jt} \\ R_{it} &= \|\mathbf{x}_i - \mathbf{x}_t\| \\ R_{it} &= \|\mathbf{x}_t\| \end{aligned} \quad (7)$$

위의 정의를 이용하여 다음과 같이 두 가지 방법으로 R_{it} 를 전개할 수 있다.

$$\begin{aligned} R_{it}^2 - (r_{ij} + R_{jt})^2 &= r_{ij}^2 + 2R_{jt}r_{ij} + R_{jt}^2 \\ R_{it}^2 - \|\mathbf{x}_i - \mathbf{x}_t\|^2 &= R_{jt}^2 - 2\mathbf{x}_t^T \mathbf{x}_i + R_{jt}^2 \end{aligned} \quad (8)$$

위의 두 식을 같다고 놓으면

$$2\mathbf{x}_t^T \mathbf{x}_i - R_{jt}^2 - r_{ij}^2 + 2R_{jt}r_{ij} + R_{jt}^2 - R_{jt}^2 = 0 \quad (9)$$

만일 i와 j의 값이 같다면 다음과 같이 식을 쓸 수 있다.

$$2\mathbf{x}_t^T \mathbf{x}_i - R_{jt}^2 + R_{jt}^2 - R_{jt}^2 = 0 \quad (10)$$

(9)식에서 (10)식을 빼면 다음과 같은 결과를 얻을 수 있다.

$$2(\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{x}_t - (R_{jt}^2 - R_{it}^2) - r_{ij}^2 - 2R_{jt}r_{ij} = 0 \quad (11)$$

이러한 벡터 형태로 나타내면 다음과 같다.

$$S_j \mathbf{x}_j = \mu_j - R_{jj} \rho_j \quad (12)$$

$$S = \begin{bmatrix} (x_2 - x_1) & (y_2 - y_1) \\ \vdots & \vdots \\ (x_N - x_1) & (y_N - y_1) \end{bmatrix}$$

$$\mu = \frac{1}{2} \begin{bmatrix} R_{20}^2 - R_{10}^2 - r_{20}^2 \\ \vdots \\ R_{N0}^2 - R_{10}^2 - r_{N0}^2 \end{bmatrix}$$

$$\rho = \begin{bmatrix} r_{21} \\ \vdots \\ r_{N1} \end{bmatrix}$$

식(12)에서는 표적과 센서의 거리에 대한 항인 R_{jk} 을 포함하고 있으나 이 값을 알 수가 없으므로 이 항을 없애기 위하여 다음과 같은 벡터를 정의한다.

$$M = (I - I^{(1)})[\text{diag}(\rho)]^{-1} \quad (13)$$

$I^{(1)}$ 은 앞에서 정의한 것과 같이 단위행렬을 오른쪽으로 회전시킨 행렬이다. M 을 식(12)식의 양쪽에 곱하면 식(14)식의 성질 때문에 식(15)식과 같이 R_{jj} 에 대한 항을 없앨 수 있다.

$$\begin{bmatrix} \text{diag}(\rho) \end{bmatrix}^{-1} \rho = \mathbf{1} \quad (14)$$

$$(I - I^{(1)}) \mathbf{1} = \mathbf{0}$$

$$M S \mathbf{x}_j = M \mu \quad (15)$$

식(15)식을 이용하여 표적의 위치 \mathbf{x}_j 를 다음과 같이 추정할 수 있다.

$$\mathbf{x}_j = (S^T M^T M S)^{-1} S^T M^T M \mu \quad (16)$$

이러한 추정방법은 다음과 같은 오차를 갖는다.[2]

$$\text{COV}[\hat{\mathbf{x}}_j] = F \Sigma F^T \quad (17)$$

$$\Sigma : r_{ij} \text{에 더해지는 잡음의 Covariance}$$

$$F = (S^T M^T M S)^{-1} S^T M^T M (\text{diag}[\rho] + R_{jj} I)$$

3.3 칼만 필터링을 이용한 smoothing

식(15)으로 추정한 표적의 위치는 센서의 위치에 의해 결정되는 기하학적인 오차를 가지기 때문에 추측이 어려운 지역이 생긴다. 이러한 점을 보완하기 위하여 식(15)의 결과를 간단한 칼만 필터링을 이용하여 smoothing시킨다.

위치 벡터는 다음과 같이 모델링할 수 있다.

$$\mathbf{x}(k) = \phi \mathbf{x}(k-1) + G w(k)$$

$$\mathbf{z}(k) = H \mathbf{x}(k) + v(k)$$

$$\mathbf{x}(k) = [x \ y \ v_x \ v_y \ a_x \ a_y]^T$$

$$\mathbf{z}(k) = [x \ y]^T$$

$$\phi = \begin{bmatrix} 1 & 0 & T & 0 & 0 & 0 \\ 0 & 1 & 0 & T & 0 & 0 \\ 0 & 0 & 1 & 0 & T & 0 \\ 0 & 0 & 0 & 1 & 0 & T \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (18)$$

$$G = [0 \ 0 \ 0 \ 0 \ 1 \ 1]^T$$

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

관측모델의 $v(k)$ 는 식(16)의 오차이므로 칼만 필터링 알고리즘에서 $\text{Cov}[v(k)]$ 는 식(17)의 값을 계산하여 사용할 수 있다.

4. Simulation

시뮬레이션 실험을 위하여 일개의 상황에 근접한 시나리오를 가상적으로 만들었다. 다음 그림 3은 6개의 소나가 국소 지역에 배치되어 있고 그 사이로 450초 동안 미지의 표적이 잠행하는 아자원 시나리오를 도식한 것이다. 표적은 20 knot(37km/h)의 속력으로 움직이며 1초 간격으로 시간 지연 차 추정치를 얻을 수 있다고 가정하였다.

시간 지연 차의 오차가 평균이 0이고 분산이 0.01인 가우시안 분포를 가진다고 가정을 하고 실험을 하였다. 그림 4는 위치 추정만을 수행한 결과이며 특정 지역에 위치 오차가 매우 심한 것을 볼 수가 있다. 그림 5와 6은 각각 시간지연차 보정과 smoothing만을 한 결과이며 그림 7은 둘 다 수행한 결과이다. 표 1은 각 실험의 오차 거리의 평균값을 보여주고 있다. 시간지연차의 보정과 smoothing을 수행함으로써 보다 정확한 위치 추적을 할 수 있음을 볼 수 있다.

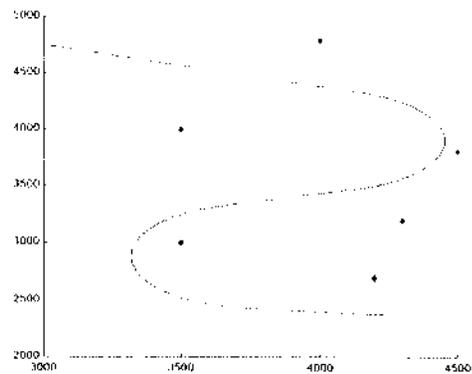


그림 3 시뮬레이션을 위한 가상 시나리오

5. 결론

본 논문에서 시간 지연 차를 이용하여 표적의 위치를 보다 정확히 추정하기 위하여 먼저 시간 지연의 차를 보정한 후 Least Square Estimation을 통하여 위치를 추적하고, 이를 칼만 필터를 이용하여 smoothing을 수행하였다.

이러한 세 단계의 과정을 거침으로써 더욱 안정적으로 위치를 추적할 수 있음을 보였다. 즉, 시간 지연 차들 사이의 관계와, 이전 위치 정보를 모두 이용하여 위치를 추정하여 추적의 질이 향상되었다.

시간 지연 차를 이용한 위치 추적은 근본적으로 시간 지연 차의 오차를 줄이는 것이 필수적이므로 정확한 시간 지연의 차를 추정할 수 있는 알고리즘이 선행되어야 한다. 또한 다중 표적의 추적을 위해서 계산량을 최소화시킬 수 있는 방법이 연구되어야 할 것이다.

Reference

- [1] R. O. Schmidt, "A New Approach to Geometry of Range Difference Location", *IEEE Transactions on Aerospace and Electronics Systems*, Vol. AES-8, No.6, pp. 821-835, Nov 1972
- [2] B. Frienlander, "A Passive Localization Algorithm and Its Accuracy Analysis", *IEEE Journal of Oceanic Engineering*, Vol. OE-12, No. 1, Jan 1987
- [3] J. O. Smith, "Closed-Form Least-Squares Source Location Estimation from Range-Difference Measurements", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-35, No. 12, Dec 1987
- [4] R. O. Schmidt, "Least Squares Range Difference Location", *IEEE Transactions on Aerospace and Electronics Systems*, Vol. 32, No. 1, Jan 1996
- [5] 암석원, 윤동현, 윤봉욱, 고한석, "미지표적의 식별과 시간지연 차의 추적연구", 1998년도 한국음향학회 하계 학술 발표회 논문집

	위치 추적	사전 보정 위치 추적	위치 추적 smoothing	사전 보정 위치 추적 smoothing
거리 오차의 평균	65.67	47.68	39.53	21.67

표 1 각 경우의 오차 거리의 평균값

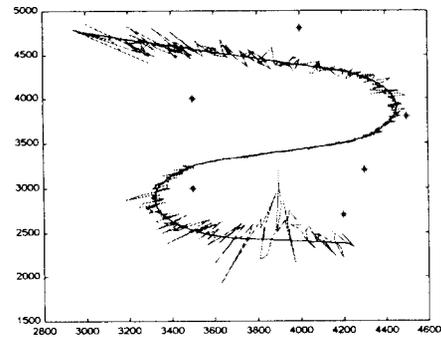


그림 4 위치 추정만 한 결과

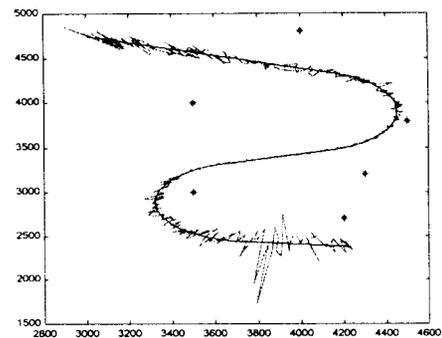


그림 5 사전 보정후 위치 추정한 결과

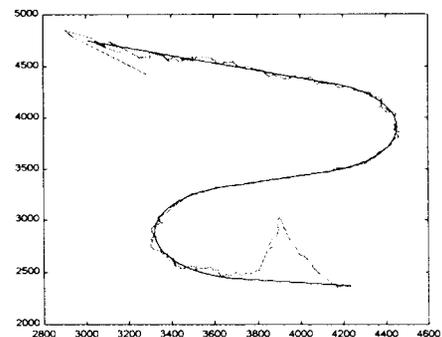


그림 6 위치 추정후 smoothing한 결과

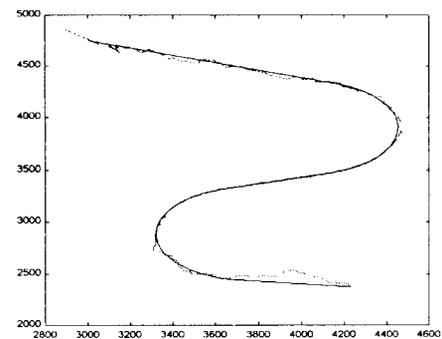


그림 7 사전보정과 smoothing을 한 결과