

등간격 평면배열 안테나에서의 Deep Neural Network 기반 방위각 및 고각 탐지

지오근, 정현진, 서형욱, 김선우
한양대학교 융합전자공학부

{wldhrms, hyeonjingoo, rlrprl, remero}@hanyang.ac.kr

Azimuth and Elevation Estimation on Uniform Planar Array using Deep Neural Network

Ohgeun Jee, Hyeonjin Chung, Hyeongwook Seo, and Sunwoo Kim
Department of Electronic Engineering, Hanyang University

요약

본 논문에서는 등간격 평면배열 안테나에서 수신 신호 방향탐지 시 보다 낮은 계산 복잡도를 요구하는 DNN (Deep Neural Network) 기반 방위각 및 고각 탐지 알고리즘을 제안한다. 제안하는 DNN 구조는 공분산 행렬의 상삼각행렬을 입력하여 추정 도래각을 출력하는 회귀 구조를 가진다. 시뮬레이션을 통해 활성화 함수에 따른 방향탐지 성능을 비교하였으며, 제안 알고리즘과 기존 방향탐지 알고리즘인 MUSIC (multiple signal classification)과의 RMSE (root mean squared error)와 실행시간을 비교한다.

I. 서론

방향탐지는 주로 레이더, 군수 통신 등에 사용되며 최근 이동통신에서의 배열 안테나 신호 처리기술이 일반화됨에 따라 민간 분야에도 확대 적용되고 있다 [1]. 하지만 하드웨어 발전에 따라 배열을 구성하는 안테나 수가 증가하는 추세이며, 이때 MUSIC (multiple signal classification)과 같은 기존 방향탐지 알고리즘 사용 시 높은 계산 복잡도가 문제가 될 수 있다 [2, 3]. 본 논문에서는 방위각 및 고각 추정 시 요구되는 계산 복잡도를 줄이고자 DNN (Deep Neural Network) 기반 방향탐지 알고리즘을 제안한다.

II. 시스템 모델

그림 1과 같이 등간격 평면배열 안테나에서 x 축 상 m 번째, y 축 상 n 번째 안테나의 좌표 $\mathbf{d}_{m,n}$ 은 $\mathbf{d}_{m,n} = [(m-1)d, (n-1)d]^T$ 를 만족하며 여기서 d 는 인접한 두 안테나 간의 거리를 의미한다. 방위각이 ϕ , 고각이 θ 일 때 배열의 x 축과 y 축에 따른 조향벡터 $\mathbf{a}(u), \mathbf{a}(v)$ 는 다음과 같이 나타낼 수 있다.

$$\mathbf{a}(u) = [1, e^{j2\pi(\frac{d\cos\phi\sin\theta}{\lambda})}, \dots, e^{j2\pi(\frac{(M-1)d\cos\phi\sin\theta}{\lambda})}]^T$$

$$\mathbf{a}(v) = [1, e^{j2\pi(\frac{dsin\phi\sin\theta}{\lambda})}, \dots, e^{j2\pi(\frac{(N-1)d\sin\phi\sin\theta}{\lambda})}]^T, \quad (1)$$

여기서 u, v 는 다음과 같이 나타낼 수 있다.

$$u = \frac{2\pi d}{\lambda} \cos\phi \sin\theta, \quad v = \frac{2\pi d}{\lambda} \sin\phi \sin\theta. \quad (2)$$

등간격 평면배열 안테나에서의 D 개의 snapshot으로 구성된 최종 수신 신호 $\mathbf{X}$ 는 다음과 같이 표현할 수 있다.

$$\mathbf{X} = \mathbf{A}\mathbf{S} + \mathbf{W} \in \mathbb{C}^{MN \times D}, \quad (3)$$

여기서 $\mathbf{W} \in \mathbb{C}^{MN \times D}$ 는 noise 행렬을 의미하고 수신 신호 행렬 $\mathbf{S}$ 는 $\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_P]^T \in \mathbb{C}^{P \times D}$ 이며, 벡터 $\mathbf{s}_i$ 는 길이가 D 인 i 번째 신호 벡터를 의미한다. $\otimes$ 는 Kronecker product를 의미한다. $\mathbf{A}$ 와 $\mathbf{a}_i$ 는 다음과 같이 나타낼 수 있다.

$$\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_P] \in \mathbb{C}^{MN \times P}, \quad (4)$$

$$\mathbf{a}_p = \mathbf{a}(u_p) \otimes \mathbf{a}(v_p) \in \mathbb{C}^{MN \times 1}. \quad (5)$$

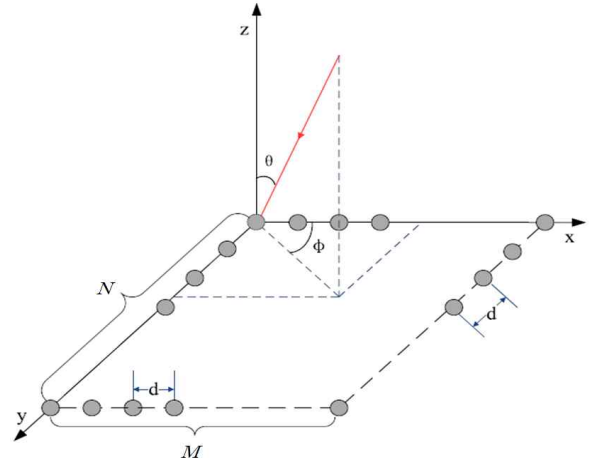


그림 1. 등간격 평면배열 안테나 구조

수신 신호 $\mathbf{X}$ 의 공분산 행렬 $\mathbf{R}_X$ 는 다음과 같이 나타낼 수 있다.

$$\mathbf{R}_X = E[\mathbf{X}\mathbf{X}^H] = \mathbf{A}\mathbf{R}_S\mathbf{A}^H + \sigma^2\mathbf{I}_{MN}$$

$$\approx \frac{\mathbf{X}\mathbf{X}^H}{D} \text{IN}C^{MN \times MN}, \quad (6)$$

여기서 $\mathbf{R}_S$ 는 신호원 $\mathbf{S}$ 의 공분산 행렬, σ 는 noise의 power, $\mathbf{I}_{MN}$ 은 $MN \times MN$ 인 단위행렬을 나타낸다. $\mathbf{R}_X$ 는 에르미트 행렬이기에, $\mathbf{R}_X$ 의 상삼각행렬을 벡터화한 $\mathbf{b}$ 는 다음과 같이 나타낼 수 있다.

$$\mathbf{b} = [R_{11}, R_{12}, R_{22}, \dots, R_{1L}, \dots, R_{LL}] \in \mathbb{C}^{(MN+1)MN/2}. \quad (7)$$

III. DNN 기반 방향탐지 알고리즘

DNN의 입력은 복소수가 될 수 없기에 (7)에서의 $\mathbf{b}$ 의 실수부와 허수부를 분리한 $\bar{\mathbf{b}}$ 와 이를 정규화한 $\mathbf{z}$ 를 입력으로 한다. $\mathbf{z}$ 는 다음과 같이 나타낼 수 있다.

$$\bar{\mathbf{b}} = [R_{11}, \text{Re}(R_{12}), \text{Im}(R_{12}), \dots, \text{Re}(R_{1L}), \text{Im}(R_{1L}), \dots, R_{LL}] \in \mathbb{R}^{(MN)^2}, \quad (8)$$

$$\mathbf{z} = \frac{\bar{\mathbf{b}}}{\|\bar{\mathbf{b}}\|} \in \mathbb{R}^{(MN)^2}. \quad (9)$$

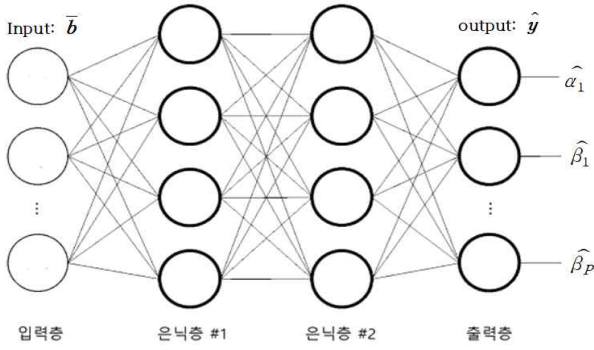


그림 2. 공분산 행렬의 상삼각행렬을 입력으로, 방위각 및 고각을 출력으로 갖는 DNN 구조

DNN의 출력 학습 데이터 $\mathbf{y}$ 는 $\mathbf{y} = [\alpha_1, \beta_1, \dots, \alpha_P, \beta_P] INR^{2P \times 1}$ 로 설정하였다. 이때, α_i, β_i 는 각각 i 번째 신호의 실제 도래각 (방위각 및 고각)이다. 이와 같이 DNN은 회귀형식으로 설정하였으며, 신호원의 개수 P 는 사전에 주어진다고 가정한다. DNN의 구조는 그림 2와 같이 도식적으로 나타낼 수 있다.

손실 함수는 MSE (mean squared error)로 설정하였으며 $\mathbf{y} - \hat{\mathbf{y}} \|^2$ 와 같이 나타낼 수 있다. 여기서 $\hat{\mathbf{y}}$ 는 신호원의 추정 도래각으로 DNN의 출력값이다. DNN의 하이퍼 파라미터는 표 1과 같다.

IV. 시뮬레이션 환경 및 결과

본 논문에서는 P 를 2로 고정하였다. 100,000개의 train data set의 SNR (signal-to-noise ratio)은 1dB부터 10dB까지 무작위로 설정하였고, RMSE 산출을 위해 총 500 test data를 사용하였다. 시뮬레이션은 Intel i7-8750H CPU, NVIDIA GeForce GTX 1050, RAM 20GB를 사용해서 진행하였다. 이외 시뮬레이션 환경 파라미터는 표 2와 같다.

표 1. DNN의 하이퍼 파라미터

하이퍼 파라미터	설정값
Activation function	ReLU, PReLU
Number of layers, nodes	4, 300
Batch size	64
Number of epochs	150

표 2. 시뮬레이션 환경 파라미터

파라미터	설정값
안테나 개수 (M, N)	5, 5
안테나 간격 (d)	0.5λ (λ : 신호의 파장)
Snapshot 개수 (D)	2048
방위각, 고각 범위	$30^\circ \leq \theta_i \leq 150^\circ, 60^\circ \leq \phi_i \leq 80^\circ$

그림 3에서 초기 epoch에서는 PReLU로 구성된 네트워크가 ReLU로 구성된 네트워크에 비해 전반적으로 적은 MSE 값을 보였으나, epoch가 거듭됨에 따라 두 DNN의 MSE가 유사해져 MSE 차이가 0.2 정도로 근소해짐을 확인하였다.

그림 4는 PReLU 사용 시 SNR에 따른 Monte Carlo 수가 500인 RMSE (root mean squared error) 결과이다. 제시한 알고리즘이 MUSIC 알고리즘에 비해 낮은 SNR에서 보다 낮은 RMSE를 가지는 것을 확인하였으며, 높은 SNR에서는 성능 차이가 좁혀지는 것을 확인하였다.

또한, 본 논문에서 제안한 DNN 기반 알고리즘이 MUSIC 알고리즘에 비해 추정시간 측면에서 장점을 가진다. MUSIC 알고리즘의 경우 각도

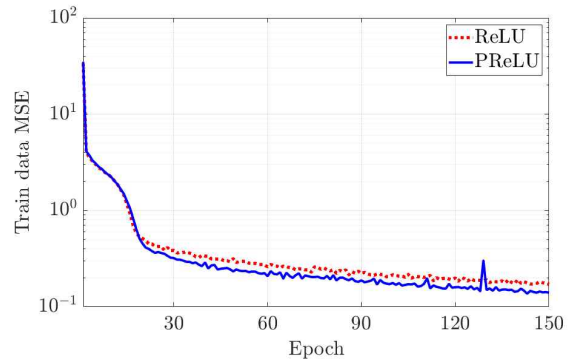


그림 3. 각 활성화 함수의 Epoch에 따른 train data MSE

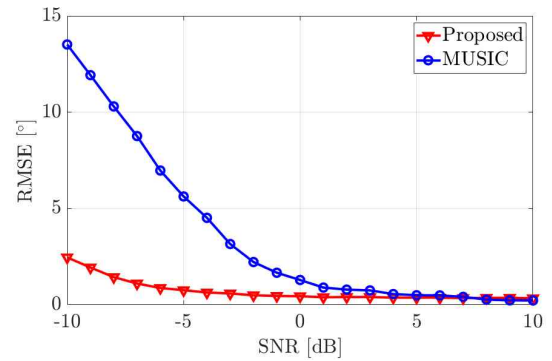


그림 4. test data의 SNR에 따른 DNN과 MUSIC 알고리즘의 RMSE 결과

스펙트럼 분할 수가 많아질수록 소요 시간이 길어지는데, MUSIC의 각도 스펙트럼을 0.5° 단위로 분해한 경우 도래각 추정에 평균 0.2초가 소요되었다. 반면, 본 논문에서 제안하는 DNN 기반 알고리즘은 평균 0.05초가 소요됨을 확인하였다.

V. 결론

본 논문에서는 DNN 기반 신호 방위각 및 고각 추정 알고리즘을 제시한다. 본 논문에서 제안하는 알고리즘은 신호의 공분산 성분을 바로 입력하여 방위각 및 고각을 출력하는 회귀 방식이다. 시뮬레이션 결과를 통해 SNR이 낮을 때 제안 알고리즘의 RMSE가 MUSIC 알고리즘보다 낮음을 확인하였으며, 실행 시간 또한 MUSIC 알고리즘에 비해 평균적으로 4배 가량 빠름을 확인하였다.

ACKNOWLEDGMENT

본 연구는 과학기술정보통신부 및 정보통신기획평가원의 대학ICT연구센터육성지원사업의 연구결과로 수행되었음 (IITP-2021-2017-0-01637)

참고 문헌

- [1] 장병준, "방향탐지기술: 튜토리얼, 최신동향 및 연구전망," *한국전자과학회논문지*, 30(8), pp. 607-618, 2019년 8월.
- [2] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Antennas, Propag.*, vol. 34, no. 3, pp. 276-280, Mar. 1986.
- [3] S. Yang and L. Hanzo, "Fifty Years of MIMO Detection: The Road to Large-Scale MIMOs," *IEEE Commun. Surveys, Tut.*, vol. 17, no. 4, pp. 1941-1988, Fourthquarter, 2015.