

FPGA에서의 Stochastic Weighted Sub-block Update (SWSU)를 이용한 MVDR 공분산 추정 가속

송원재, 이성주*

세종대학교 반도체시스템공학과 및 지능형드론융합전공,
*세종대학교 AI 융합 전자공학과 및 지능형드론융합전공

wonjae@itsoc.sejong.ac.kr, *seongjoo@sejong.ac.kr

Accelerating MVDR Covariance Estimation on FPGA Using Stochastic Weighted Sub-block Update (SWSU)

Wonjae Song, Seongjoo Lee*

Dept. of Semiconductor Systems Engineering and Dept. of Convergence Engineering for Intelligent Drone, Sejong Univ., *Dept. of AI Convergence Electronic Engineering and Dept. of Convergence Engineering for Intelligent Drong, Sejong Univ.

요약

본 논문에서는 MVDR(Minimum Variance Distortionless Response) 범포밍의 핵심 연산인 공분산 행렬 역행렬 계산의 실시간 처리 한계를 극복하기 위해, Stochastic Weighted Sub-block Update(SWSU) 기법을 제안한다. 제안 기법은 연산 복잡도를 평균 $O((1-\tau)\cdot((M/P)^2))$ 로 감소시킨다. MATLAB 기반 시뮬레이션 결과, 제안 SWSU MVDR는 기존 MVDR 대비 약 3배 빠른 처리 속도와 1.12배 수준의 오차율을 달성하였으며, FPGA 구현에서는 LUT 6%, FF 18% 감소 및 URAM 제거, 전력 소모 약 7% 절감을 입증하였다. 이를 통해 SWSU 기법이 고채널 수 환경에서 실시간 하드웨어 범포밍에 적합함을 확인하였다.

I. 서 론

MVDR(Minimum Variance Distortionless Response) 범포밍은 간접 악제 및 신호 증폭 성능이 뛰어나 다양한 레이더·통신 시스템에서 널리 사용된다. 그러나 MVDR의 핵심 연산인 공분산 행렬 R 의 추정 및 역행렬 계산은 $O(M^3)$ 의 계산 복잡도를 가지며^[1], 채널 수 M 이 커질수록 실시간 구현이 매우 어려워진다. 이를 완화하기 위해 본 논문은 확률적(stochastic) 또는 랜덤화(randomized) 업데이트를 도입하여,^{[2][3]} 매 샘플마다 업데이트 강도나 수행 여부를 난수에 기반해 결정함으로써 연산량을 동적으로 조절하는 “Stochastic Weighted Sub-block Update (SWSU)” 알고리즘을 제안하고 해당 SWSU 알고리즘을 MATLAB과 FPGA에 구현해 기존의 MVDR 범포밍과 비교한다.

II. 본론

1. 서브 블록 분할 전략

전체 공분산 행렬 $R \in C^{M \times M}$ 을 p 개의 서브 블록으로 분할한다. 보통 행 단위로 균등 분할하여, 각 블록 크기가 $M/P \times M$ 인 형태로 나누거나, 대각 블록 형태로 $M/P \times M/P$ 크기로 분할할 수 있다. 제안 알고리즘에서는 후자를 선택하며, 블록 인덱스 $b=1, 2, \dots, p$ 에 대응하는 부분 행렬을 식 (1)로 정의한다.

$$R_b = R \left[(b-1) \frac{M}{P} + 1 : b \frac{M}{P}, (b-1) \frac{M}{P} + 1 : b \frac{M}{P} \right] \quad (1)$$

$$\alpha < U(0,1), \quad \tau \in [0,1] \text{ (threshold)} \quad (2)$$

$$R_b^{-1}(n) = R_b^{-1}(n-1) - \frac{R_b^{-1}(n-1)(\sqrt{\alpha}x_{n,b})(\sqrt{\alpha}x_{n,b})^H R_b^{-1}(n-1)}{1 + (\sqrt{\alpha}x_{n,b})^H R_b^{-1}(n-1)(\sqrt{\alpha}x_{n,b})} \quad (3)$$

$$w_{MVDR} = \frac{R^{-1}(n)\alpha(\theta_0)}{\alpha^H(\theta_0)R^{-1}(n)\alpha(\theta_0)} \quad (4)$$

2. 확률적 가중 업데이트

매 입력 샘플 $x_n \in C^M$ 이 들어올 때마다, 라운드 로빈 방식으로 블록 b 하나를 지정해 서브 블록을 선택한다. 식 (2)에 따라 확률적 가중치를 산출한다. 이때 $\alpha < \tau$ 이면 업데이트를 완전히 건너뛰고, 그렇지 않으면 부분 업데이트를 수행한다. $x_{n,b}$ =블록 b 에 해당하는 부분 벡터를 취하여, (Rank Update) sherman-morrison 기반의 식 (3)으로 부분 업데이트를 한다. 이때 전체 행렬 R^{-1} 은 업데이트된 블록만 교체하고 나머지 블록을 그대로 유지한다.

3. 연산 복잡도 분석

기존의 MVDR의 핵심 연산인 공분산 행렬 R 의 추정 및 역행렬 계산은 $O(M^3)$ 의 계산 복잡도를 가진다고 언급했다. 본 논문은 서브 블록 업데이트

*교신저자

이트 방식을 채택해 한 블록의 크기가 $M/P * M/P$ 가 된다. 식 (3)은 분모 연산에서 $O(M^2)$, 문자 연산에도 $O(M^2)$ 복잡도를 가져 식 (3)의 연산 복잡도는 $O(M^2)$ 가 되므로, 연산 복잡도는 $O((M/p)^2)$ 가 된다.^[4] 또한 확률적 스킵으로 평균 업데이트 확률의 $(1-\tau)$ 만큼 수행하므로 본 논문의 연산 복잡도는 $(1-\tau)O((M/p)^2)$ 가 된다. 이를 통해 p 와 τ 를 적절히 설정하면, 기존 대비 최대 *배 및 스킵률에 따른 추가 절감이 가능하다.^[5]

4. MVDR 베포밍 가중치 계산과 통합

SWSU로 개신된 역공분산 행렬을 사용하여, 전형적인 MVDR 가중치를 식 (4)를 이용해 계산한다. 서브 블록별로 개신된 부분만 최신 정보가 반영되므로, 전체 가중치 계산에서도 부분 업데이트된 블록^[6] 기억을 적절히 통합하여 성능 저하를 최소화한다.

5. 실험 결과

MATLAB 시뮬레이션에서 기준의 MVDR 베포밍이 83.070(s), SWSU MVDR 베포밍이 27.959(s)로 약 3배 빠른 시뮬레이션 속도를 보여줬다. 그에 비해 오차율은 표 1에서 볼 수 있듯 약 1.1배 정도만 증가했다. FPGA에서 구현한 결과는 기준 MVDR 베포밍이 1002.62(W), SWSU MVDR 베포밍이 935.40(W)로 전력이 약 0.9배 하락했고, 자원 사용량은 표 2를 보면 DSP를 제외하고 모두 크게 감소한 것을 확인할 수 있다.

표 1. MVDR 베포밍 오차율 비교

Table 1. MVDR Beamforming Error Rate Comparison

	Conventional MVDR Beamforming	SWSU MVADR Beamforming
Normalization magnitude max error	0.1227	0.1377
Normalization magnitude inverted	2.432e-04	2.731e-04

표 2. FPGA 자원 사용량 비교

Table 2. FPGA Resource Utilization Comparison

	Conventional MVDR beamforming	SWSU MVDR beamforming
LUT	36758	34631
FF	31037	25644
BRAM	4	4
URAM	16	0
DSP	106	116

III. 결론

본 논문에서 제안한 Stochastic Weighted Sub-block Update(SWSU) 알고리즘은 공분산 행렬을 p 개의 서브 블록으로 분할하고 확률적 가중(α)과 스kip 임계값(τ)을 결합함으로써 연산 복잡도를 평균 $O((1-\tau)(M/p)^2)$

수준으로 낮추었으며, MATLAB 시뮬레이션에서 기준 MVDR 대비 약 3배 빠른 처리 속도와 1.12배 수준의 오차율 증가를, FPGA 구현에서는 LUT 사용량을 약 6%, FF 사용량을 약 18% 절감하고 URAM을 제거하며 전력 소모를 약 7% 절감함으로써 실시간 하드웨어 환경에서 고효율 MVDR 베포밍 구현의 가능성을 입증하였다.

ACKNOWLEDGMENT

본 연구는 과학기술정보통신부의 재원으로 한국연구재단, 무인이동체원천기술개발사업단의 지원을 받아 무인이동체원천기술개발사업과(No. 2023M3C1C1A01098414) 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행되었고(No. 2023R1A2C1006340) 검증을 위한 EDA관련 툴은 IDEC의 지원을 받았음.

참 고 문 현

- [1] Z. A. Shubber, T. M. Jamel and A. K. Nahar, "Performance Evaluation of Beamforming Array Antennas Based on Partial Update LMS and NLMS Adaptive Algorithms," 2023 6th International Conference on Engineering Technology and its Applications (IICETA), Al-Najaf, Iraq, 2023, pp. 708-713.
- [2] Y.-H. Choi, "Adaptive Beamforming Robust to Pointing Errors Through Extraction of Interference Correlation Matrix," J. Korean Inst. Commun. Inf. Sci., vol. 48, no. 7, pp. 795 - 802, Jul. 2023.
- [3] S.-H. Liu, C.-Y. Kuo, Y.-N. Mo and T. Su, "An area-efficient, conflict-free and configurable architecture for accelerating NTT/INTT," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 32, no. 3, pp. 519 - 533, Mar. 2024.
- [4] Y. Huang, S. A. Vorobyov and Z. -Q. Luo, "Quadratic Matrix Inequality Approach to Robust Adaptive Beamforming for General-Rank Signal Model," in IEEE Transactions on Signal Processing, vol. 68, pp. 2244-2255, 2020.
- [5] S. Vadhvana, S. K. Yadav, S. S. Bhattacharjee and N. V. George, "An Improved Constrained LMS Algorithm for Fast Adaptive Beamforming Based on a Low Rank Approximation," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 69, no.8, pp. 3605-3609, Aug. 2022.