

Slotted-ALOHA 시스템에서 스파이킹 신경망 기반 백오프 기법

강민정*, 이정훈*, 정동근^o

Spiking Neural Network-Based Backoff Scheme for Slotted-ALOHA Systems

Min Jeong Kang*, Jung Hoon Lee*,
Dong Geun Jeong^o

요약

본 레터에서는 Slotted-ALOHA 시스템의 성능을 높이기 위해 상대적으로 저전력으로 동작하며 구현 복잡도가 낮은 스파이킹 신경망(spiking neural network, SNN)을 활용한 백오프(backoff) 기법을 제안한다. 제안하는 기법에서 SNN 모듈은 이진 지수 백오프(binary exponential backoff, BEB) 기법으로 동작하는 시스템의 전송 성공, 전송 실패 등의 관찰 데이터를 실시간으로 입력받아 최적의 경쟁 윈도우 값을 찾는 다. 시뮬레이션을 통하여 제안하는 기법의 성능을 평가하였으며, 성능 평가를 통해 제안하는 기법이 기존의 BEB 기법에 비해 더 높은 성능을 가지며, 수치적으로 찾은 최적과 근접한 성능을 가짐을 확인하였다.

Key Words : Backoff scheme, machine learning, slotted-ALOHA, spiking neural network

ABSTRACT

In this letter, we propose a spiking neural network (SNN)-based backoff scheme to enhance the performance of slotted-ALOHA systems, which operates with relatively low power and implementation complexity. In our scheme, an SNN model takes real-time input of observation data, such as transmission success and failure, from the system operating with binary exponential backoff (BEB), and returns the optimal contention window size. Through simulations, we show that our proposed scheme achieves higher performance compared to the traditional BEB, approaching the performance of the optimal backoff scheme numerically optimized.

1. 서론

Slotted-ALOHA는 단순한 동작 방식과 낮은 전력 소비 등의 장점으로 인하여 저전력 사물인터넷(Internet of Things, IoT) 기기들의 랜덤 액세스 환경에서 널리 사용되고 있다. Slotted-ALOHA에서는 다수의 기기들이 보내는 패킷(packet) 간의 충돌(collision)을 감소시키기 위해 경쟁 윈도우(contention window, CW) 안에서 임의의 시간 동안 대기 후 패킷을 전송하는 백오프(backoff) 기법을 사용한다. 이에 따라 최적의 CW 값을 도출해내는 것이 성능에 직접적인 영향을 미치며, 전송 실패 시 두 배씩 CW 값을 늘려서 사용하는 기존의 이진 지수 백오프(binary exponential backoff, BEB) 기법의 성능을 향상시키기 위한 다양한 연구가 진행되었다^{1,2}.

기계학습(machine learning, ML) 기술의 발전에 따라 무선 통신 시스템의 성능 향상을 위해 다양한 기계학습 모델을 활용하는 연구가 활발히 이루어지고 있다. 기계학습 모델은 복잡한 함수의 범용 근사화기(universal approximator)로서 한편으로는 복잡한 문제를 낮은 복잡도로 해결하는 것을 가능하게 한다. 하지만

* This work was supported by the Hankuk University of Foreign Studies Research Fund and by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2022R1A2C1004928).

• First Author : (ORCID:0000-0001-7714-1504) Department of Electronics Engineering and Applied Communications Research Center, Hankuk University of Foreign Studies, love_minmin926@hufs.ac.kr, 학생(박사과정), 정회원

◦ Corresponding Author : (ORCID:0000-0001-7515-7422) Department of Electronics Engineering and Applied Communications Research Center, Hankuk University of Foreign Studies, dgjeong@hufs.ac.kr, 정교수, 종신회원

* (ORCID:0000-0002-7875-2398) Department of Electronics Engineering and Applied Communications Research Center, Hankuk University of Foreign Studies, tantheta@hufs.ac.kr, 부교수, 정회원

논문번호 : 202312-171-B-LU, Received December 23, 2023; Revised January 2, 2024; Accepted January 2, 2024

높은 정확도를 얻는 데 필요한 학습 데이터의 양이 많아 질수록 하드웨어가 소모하는 전력량이 크므로 전력이 제한된 환경에서는 사용하기가 쉽지 않다. 최근 스파이킹 신경망(spiking neural network, SNN)은 제 3세대 인공신경망(artificial neural network, ANN) 기술로 많은 주목을 받고 있다. SNN은 생물학적인 뉴런의 신호 전달 기작과 매우 유사하게 동작하여 회로 구현에 용이하고, 뉴런 간에 이루어지는 정보 전달에 스파이크(spikes) 신호를 사용하여 전력 소비량이 낮은 기계학습 모델이다³⁾. 이에 따라, 기계학습 모델을 기반으로 하는 시스템에 기존 기계학습 모델 대신 SNN을 사용하여 소모 전력을 감소시키는 연구가 활발히 이루어지고 있다⁴⁾.

본 레터에서는 Slotted-ALOHA 시스템의 성능 향상을 위해 낮은 전력으로 동작하고 회로 구현이 용이한 SNN 기반 백오프 기법을 제안한다. 제안하는 기법은 BEB 기법으로 동작하는 Slotted-ALOHA 시스템의 중앙제어기(e.g., LoRa의 게이트웨이, 센서 네트워크의 수집기 등)에서 전송 성공 및 전송 실패 관찰 데이터를 실시간으로 SNN 모듈에 입력하여 최적의 CW 값을 찾은 후, 이를 고정하여 사용하는 기법이다. 시뮬레이션을 통하여 제안하는 기법의 성능이 기존의 기법의 성능을 크게 향상시키는 것을 보인다.

II. 시스템 모델

본 레터에서는 Slotted-ALOHA를 사용하는 IoT 네트워크에서 총 N 개의 IoT 기기가 하나의 중앙제어기에 패킷을 보내는 상황을 고려한다. 모든 기기들은 동일한 길이의 패킷을 전송한다고 가정하며, 각 기기는 전송할 패킷이 생성된 경우 이를 송신 큐에 저장하고, 전송할 순서가 되면 해당 슬롯(slot)의 시작 지점에서 패킷을 전송한다. 한편, 각 슬롯의 상태는 해당 슬롯에서 몇 명의 기기가 패킷을 전송했는지에 따라 1)성공(한 기기만 전송), 2)충돌(두 기기 이상 전송), 3)휴면(아무 기기도 전송 안함)의 세 가지 상태로 결정된다. 패킷 간 충돌을 방지하기 위해 각 기기는 CW 값을 사용하여 $[0, CW - 1]$ 구간 내의 균등(uniform)분포에서 추출한 난수 슬롯 수만큼 대기한 후 전송한다. 따라서 각 기기의 CW 조절 방법은 패킷 충돌 확률과 직접적으로 연관되며, 시스템 성능에 큰 영향을 미친다.

기존에 널리 사용되는 BEB의 경우 CW 값은 시스템에서 정해진 최소 CW 값(CW_{min})으로부터 시작하여 전송에 실패할 경우 두 배씩 증가하며, 최대 CW 값

(CW_{max})으로 제한된다. 따라서 한 기기에서 n 번 전송 실패 후 얻는 CW 값은 다음과 같다.

$$CW = \min[CW_{min} \times 2^n, CW_{max}]$$

전송에 성공할 경우, CW 값은 CW_{min} 값으로 초기화되며, 미리 정해진 최대 재전송 횟수를 초과할 경우, 해당 패킷은 폐기되고 기기는 다음 패킷 전송을 준비한다.

III. 스파이킹 신경망 기반 백오프 기법

3.1 스파이킹 신경망

SNN은 뉴런의 전위가 정해진 막 전위 임계치 이상이 될 경우, 스파이크 형태의 신호를 형성해 다음 뉴런에게 신호를 전달한다는 점에서 생물학적인 뉴런의 신호 전달 기작과 높은 유사도를 갖는 기계학습 모델이다. 특히, 각각의 뉴런은 저전력의 스파이크를 사용하여 개별적으로 학습을 수행하고, 백터 연산과 같이 복잡한 연산을 수행하지 않는다는 점에서 심층 신경망(deep neural network, DNN)과 합성곱 신경망(convolutional neural network, CNN)보다 상대적으로 낮은 전력과 적은 계산량을 필요로 한다⁵⁾.

전체적인 SNN의 구조는 기존의 ANN과 유사하며, 각 노드에 사용되는 인공 뉴런(퍼셉트론)의 형태만 차이를 보인다. 그림 1은 두 개의 입력 데이터를 받아 하나의 출력 데이터를 생성하는 간단한 인공 스파이킹 뉴런을 나타낸다. 이때, 인공 스파이킹 뉴런의 전위는 시냅스 이전 뉴런이 보내오는 스파이크 신호가 입력되는 시간 간격에 영향을 받는다.

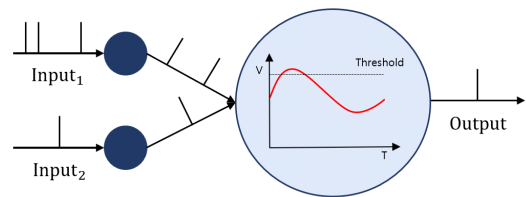


그림 1. 일반적인 인공 스파이킹 뉴런
Fig. 1. Conventional artificial spiking neuron

3.2 제안하는 스파이킹 신경망 기반 백오프 기법

본 논문에서 제안하는 SNN 기반 백오프 기법은 총 세 단계로 동작한다. 1)관찰 단계, 2)방송 단계, 3)고정 CW 사용 단계. 먼저 1)관찰 단계에서는 모든 기기들이 T 슬롯 동안 BEB 기법으로 동작하며, 이때 중앙제어기

는 각 기기들의 접속 성공 또는 전송 실패 데이터를 SNN 모듈에 실시간으로 입력한다. 관찰 시간이 지난 후 SNN 모듈로부터 반환된 최적으로 예측되는 CW 값을 얻는다. 2)방송 단계에서는 한 슬롯을 사용하여 중앙 제어기가 SNN에서 찾은 CW 값을 모든 기기들에게 알린다. 3)고정 CW 사용 단계에서는 기기들이 방송 단계에서 전달 받은 CW 값을 고정하여 사용한다.

제안하는 기법에서 사용하는 SNN 모듈의 구조는 그림 2와 같다. 최대 N 개의 기기가 존재할 때, 제안하는 SNN 모듈은 $N+1$ 개의 입력 노드를 가진다. 만약 n 번째 기기가 전송에 성공하면 n 번째 입력 노드에 스파이크가 실시간으로 입력되고, 전송 실패 시 $N+1$ 번째 입력 노드에 스파이크가 실시간으로 입력된다. 제안하는 SNN 구조는 M 개의 출력 노드를 가지며, 각 출력 노드는 서로 다른 M 개의 고정된 CW 값을 나타낸다. 관찰 시간인 T 슬롯 동안 실시간으로 스파이크 입력이 들어간 후, 최적으로 예측되는 CW 값에 해당하는 출력 노드로 스파이크를 출력한다. 한편 SNN 모듈의 복잡도를 더욱 줄이기 위해 SNN 모듈의 내부에서 계수기(counter)를 사용할 수 있다. 계수기는 미리 설정된 α 슬롯 구간마다 각 기기가 성공한 전송 성공 횟수와, 전송 실패 횟수를 각각 계수하여, 구간이 끝나는 시점에서 각 횟수에 비례한 크기의 스파이크들을 다음 노드에 전달한다.

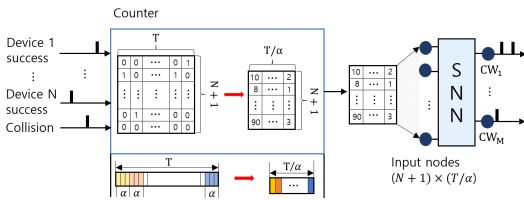


그림 2. 제안하는 SNN 모듈
Fig. 2. Proposed SNN module

IV. 성능 평가

제안하는 기법의 성능을 평가하기 위해 총 70개의 기기가 존재하는 Slotted-ALOHA 환경을 고려한다 ($N = 70$). 제안하는 기법은 10000 슬롯 시간 동안 관찰 단계로 동작한 후, 한 슬롯 동안 방송 단계로 동작하며, 이후 10000 슬롯 동안 고정 CW 단계로 동작하도록 설정하였다. 사용한 SNN 모듈의 구조는 $71 (= N + 1)$ 개의 입력과, 128개의 은닉 노드를 가진 1개의 은닉 층을 가지도록 설계하였다. 또한, BEB 기법에서 사용하는 7개의 CW 값 {4, 8, 16, 32, 64, 128, 256} 중 하나를 고정하여 사용할 수 있도록 각 CW의 값에 해당하는

7개의 출력 노드를 가지도록 설계하였다. 한편 SNN 모듈의 복잡도를 줄이기 위해 100 슬롯 간격으로 입력을 증가하도록 계수기를 설정하였다($\alpha = 100$). SNN 모듈은 Keras 프레임워크 내에서 제공하는 KerasSpiking을 사용하여 학습하였으며, 각 은닉 노드와 출력 노드의 활성화 함수로 Rectified linear unit(ReLU) 함수를 사용하였다. 이때, 손실 함수와 최적화 알고리즘으로 각각 categorical cross entropy와 Adaptive moment(Adam) 알고리즘을 사용하였다. 최대 재전송 횟수는 6회로 가정하였다.

그림 3과 그림 4는 각각 한 기기가 한 슬롯 동안 평균적으로 생성하는 패킷 수에 따른 충돌 확률과 시스템 처리량을 나타낸다. 각 기기에서 생성되는 패킷은 포아송 분포를 따른다고 가정하였으며, 성능평가를 위해 총 200010 슬롯 시간 동안 시뮬레이션을 진행하였다. 그림

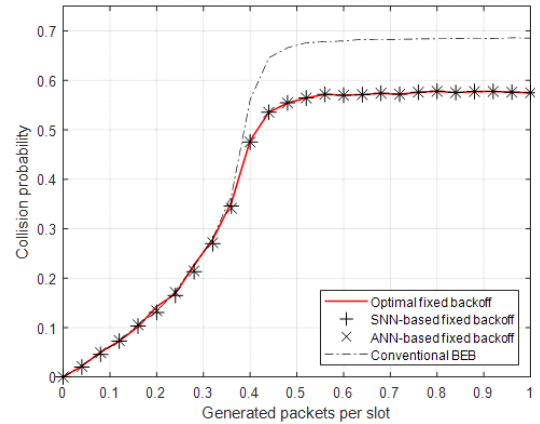


그림 3. 각 기기 트래픽에 따른 충돌 확률
Fig. 3. The collision probability with respect to device traffic

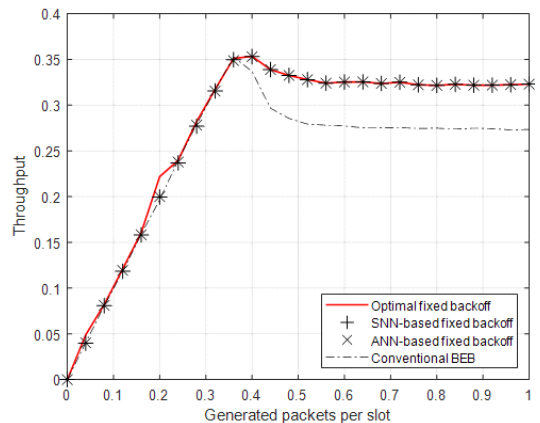


그림 4. 각 기기 트래픽에 따른 시스템 처리량
Fig. 4. System throughput with respect to device traffic

3과 그림 4를 통해 슬롯 시간당 각 기기들의 트래픽이 증가함에 따라 제안하는 기법이 기존 BEB 기법보다 높은 성능을 가짐을 확인할 수 있다. 또한 제안하는 기법이 수치적으로 찾은 최적의 기법과 ANN을 사용한 기법과 거의 유사한 성능을 가짐을 확인할 수 있다.

V. 결 론

본 레터에서는 Slotted-ALOHA 시스템 성능을 높이기 위해 낮은 전력으로 동작하고 회로 구현이 용이한 SNN을 활용하는 백오프 기법을 제안하였다. 성능평가를 통해 제안하는 기법을 사용할 경우, 기존의 BEB 기법보다 높은 성능을 얻을 수 있음을 확인하였으며, 수치적으로 찾은 최적의 기법과 유사한 성능을 가지는 것을 확인하였다.

References

- [1] S. Chun, D. Xianhua, L. Pingyuan, and Z. Han, "Adaptive access mechanism with optimal contention window based on node number estimation using multiple thresholds," *IEEE Trans. Wireless Commun.*, vol. 11, no. 6, pp. 2046-2055, June. 2012. (<https://doi.org/10.1109/TWC.2012.040412.110080>)
- [2] H. J. Kwon, J. H. Lee, and D. G. Jeong, "Machine learning based backoff scheme for Slotted-ALOHA," *J. KICS*, vol. 45, no. 1, pp. 34-37, Jan. 2020. (<https://doi.org/10.7840/kics.2020.45.1.34>)
- [3] A. Tavanaei, M. Ghodrati, S. R. Kheradpisheh, T. Masquelier, and A. Maida, "Deep learning in spiking neural networks," *Neural Netw.*, vol. 111, pp. 47-63, Mar. 2019. (<https://doi.org/10.1016/j.neunet.2018.12.002>)
- [4] R. Mao, et al., "An ultra-energy-efficient and high accuracy ECG classification processor with SNN inference assisted by on-chip ANN learning," *IEEE Trans. Biomed. Circuits Syst.*, vol. 16, no. 5, pp. 832-841, Oct. 2022. (<https://doi.org/10.1109/TBCAS.2022.3185720>)