

# 임베디드 시스템에서의 양자화 기계학습을 위한 양자화 오차보상에 관한 연구

석 진욱, 김 정시

한국전자통신연구원 인공지능연구소 초성능컴퓨팅연구본부 고성능디바이스 SW 연구실  
jnwseok@etri.re.kr, sikim00@etri.re.kr

## Study on Quantized Learning for Machine Learning Equation in an Embedded System

Jinwuk Seok, Jeong-Si Kim

Electronics and Telecommunications Research Institute, Artificial Intelligence Research Lab., Future Computing Research Division, High Performance Embedded System SW Research Section

### 요 약

본 논문에서는 임베디드 시스템에서의 양자화 기계학습을 수행할 경우 발생하는 양자화 오차를 효과적으로 보상하기 위한 방법론을 제안한다. 경사 도함수(Gradient)를 사용하는 기계학습이나 비선형 신호처리 알고리즘에서 양자화 오차는 경사 도함수의 조기 소산(Early Vanishing Gradient)을 야기하여 전체적인 알고리즘의 성능 하락을 가져온다. 이를 보상하기 위하여 경사 도함수의 최대 성분에 대하여 직교하는 방향의 보상 탐색 벡터를 유도하여 양자화 오차로 인한 성능 하락을 보상하도록 한다. 또한, 기존의 고정 학습률 대신, 내부 순환(Inner Loop) 없는 비선형 최적화 알고리즘에 기반한 적응형 학습률 결정 알고리즘을 제안한다. 실험 결과 제안한 방식의 알고리즘을 비선형 최적화 문제에 적용할 시 양자화 오차로 인한 성능 하락을 최소화시킬 수 있음을 확인하였다.

### 1. 서론

기존의 기계학습 혹은 비선형 신호 처리의 경우, 부동 소수점 연산을 기반으로 연산을 수행한다. 그러나, 기계학습과 같이 다수의 연산 모듈을 사용하여 실시간성을 추구하는 대상의 경우 부동 소수점 계산을 위한 연산 모듈의 크기와 복잡도가 정수 연산보다 상대적으로 크기 때문에 소형, 경량의 하드웨어를 필요로 하는 분야에서는 적합하지 않다[1] [2].

따라서, 비트 감소, 계산 속도 개선, 가용성 확장과 같은 엔지니어링 측면의 장점을 제공해 줄 수 있는 처리 데이터의 양자화는 다양한 엔지니어링 분야에서 연구되었다. 통상적인 경우, 양자화 된 도메인의 학습 방정식은 파라미터의 최하위 비트의 업데이트로 구현되므로, 업데이트 파라미터에 일정한 학습 속도를 적용하는 것과 동일하다. 그러나 단계 크기가 일정한 일반적인 확률론적 최급 강하 알고리즘은 1 차 분포와의 수렴과 같이 매우 약한 토폴로지서 최적의 점으로 수렴하기에 성능 하락을 피할 수 없다[3] [4] [5].

본 논문에서는 이와 같은 양자화의 이점을 기계학습에 인식하기 위해, 양자화로 인한 성능하락을 극복하기 위한 적응적 학습률의 기계학습 적용 알고리즘을 제안하고, 이를 기반으로 양자화 오차 보상을 위한 보상 알고리즘을 제안하며, 이를 강화학습 등의 기계학습에 적용 가능한 비선형 최적화 문제에 적용하여 제안한 양자화 오차 보상 방법론이 타당함을 보인다.

본 논문의 구성은 다음과 같다. 2 절에서는 양자화 및 주요 양자화 연산에 대한 정의를 두고, 3 절에서는 양자화 기반 최급 강하법의 정의와 적응적 학습률의 기계학습 정의를 위한 알고리즘을 제안하고 4 절에서는 양자화 기계학습시 발생하는 양자화 오차 보상 알고리즘을 설명한다. 5 절에서는 제안한 기법의 성능을 실험을 통해서 확인하며, 마지막으로 6 절에서는 본 논문에 대한 결론을 맺는다.

### 2. 양자화 및 주요 양자화 연산 정의

먼저 변수  $x \in \mathbf{R}$ 의 양자화의 정의를 위해 다음과 같이 정수화를 위한 아래 자리 수 버림을 (Round-off) 를 다음과 같이 정의한다[3].

$$x \equiv [x] + \epsilon \quad (\epsilon \in \mathbf{R}[0,1]) \quad (1)$$

식 (1)에서 기호  $[x] \in \mathbf{Z}$  는 자리 버림 연산을 나타내는 것으로서  $x$  보다 작은 정수 값 중 가장 가까운 값을 가지는 것으로 정의한다. 이를 사용하여 다음과 같이 가우스 기호 (Gauss symbol)를 다음과 같이 정의하자

$$[x] \equiv [x + 0.5] = x + 0.5 - \epsilon \triangleq x + \epsilon \quad (2)$$

이를 사용하여  $x$ 의 반올림 연산을  $[x]$ 으로 정의하면, 이것의 자리 오차 (round-off error)  $\epsilon$ 는  $\epsilon \in \mathbf{R}(-0.5,0.5)$ 로 정의된다. 또한, 임의의 실수 열  $x_k, \forall k \in \mathbf{N}, x_k \in \mathbf{R}$ 에 대하여 식 (1)과

같이 각  $x_k$  에 대한 자리 버림을  $\epsilon_k$  로 나타낼 수 있다고 하면

$$\left| \sum_{k=1}^n x_k \right| = \left| \sum_{k=1}^n (x_k + \epsilon_k) \right| = \sum_{k=1}^n |x_k| + \left| \sum_{k=1}^n \epsilon_k \right| \quad (3)$$

이에, 식 (1),(2)의 정의를 사용하여 다음과 같이 양자화 연산을 정의한다

$$x^Q \triangleq \frac{1}{Q_p} [Q_p \cdot (x + 0.5 \cdot Q_p^{-1})] \quad (4)$$

식 (4)에서  $Q_p$  는 양자화 계수로서 양자화의 수준을 결정한다. 예를 들어,  $10^{-3}$  수준의 고정 소수점으로 양자화를 수행하려고 한다면  $Q_p = 10^3$  로 결정된다. 편의상 양자화 계수는 양의 정수로 놓는다 ( $Q_p \in \mathbf{Z}, Q_p > 0$ ). 특정 정수의 가우스 기호 없이 양자화를 표현하기 위하여 자리 오차를 도입하여 식 (4)를 다시 쓰면 다음과 같다.

$$\begin{aligned} x^Q &= \frac{1}{Q_p} [Q_p \cdot x] = \frac{1}{Q_p} [Q_p \cdot (x + 0.5Q_p^{-1})] \\ &= \frac{1}{Q_p} (Q_p \cdot x + \epsilon) = x + \epsilon Q_p^{-1} \end{aligned} \quad (5)$$

식 (5)에서  $x^Q \in \mathbf{R}$  이지만,  $Q_p x^Q = [Q_p \cdot x] \in \mathbf{Z}$  이다. 따라서  $x^Q$  는 단순히 양자화가 정수화를 의미하는 것이 아닌 고정 소수점의 형태까지 의미하게 되며 자리오차  $\epsilon$ 의 범위는  $\epsilon \in \mathbf{R}(-0.5Q_p^{-1}, 0.5Q_p^{-1}) = \mathbf{R}(-5 \cdot (10Q_p)^{-1}, 5 \cdot (10Q_p)^{-1})$  이 된다. 만일,  $Q_p$  가 충분히 커서  $Q_p$ 에 따른 자리 오차의 분포에 대한 평균 값  $\mathbb{E}_{Q_p} \epsilon = 0$  이 되도록 하면,  $x$ 의 샘플 데이터  $x_k$ 에 대하여, 다음의 관계가 성립한다.

$$\begin{aligned} \mathbb{E}_{Q_p} x^Q &\triangleq \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=0}^{N-1} x_k^Q = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=0}^{N-1} (x_k + \epsilon_k Q_p^{-1}) \\ &= \mathbb{E}_{Q_p} x + Q_p^{-1} \mathbb{E}_{Q_p} \epsilon = \mathbb{E}_{Q_p} x. \end{aligned} \quad (6)$$

기계학습의 학습 방정식에 양자화를 적용하게 되는 경우를 분석하자, 먼저,  $x(t) \in \mathbf{R}^n$ 에 대하여 다음과 같은 형태의 학습 방정식이 주어졌다고 가정한다.

$$x_{t+1} = x_t - \lambda_t h_t \quad (7)$$

식 (7)에서  $h_t \in \mathbf{R}^n$ 은 탐색방향 벡터이고  $\lambda_t \in \mathbf{R}(0,1]$ 은 학습률이며  $t \in \mathbf{R}$ 는 시간에 대한 파라미터이다. 식 (7)에서  $x_t$ 의 양자화를  $x_t^Q \equiv [x_t]$ 로 정의하면 다음과 같은 양자화된 학습 방정식을 얻게 된다.

$$x_{t+1}^Q = (x_t^Q - (\lambda_t h_t)^Q)^Q = x_t^Q - (\lambda_t h_t)^Q \quad (8)$$

학습률  $\lambda_t \in \mathbf{R}$ 와 탐색 방향 벡터  $h_t \in \mathbf{R}^n$ 가 각각 스칼라 및 벡터이므로,  $\lambda_t h_t = (\lambda_t h_t)^Q$ 로 만드는 양자화는 구현이 어렵다. 그러므로, 최적의 학습률을 선 탐색(Line Search) 알고리즘을 통해 구하더라도, 이를 양자화가 가능하도록 다시 계산해 주어야 한다. 만일 단위 양자화 탐색 방향 벡터가 유리수로 잘 정의되었다면 (i.e.  $\frac{1}{Q_p} h_t \in \mathbf{Q}^n$ ),  $t$ 번째 전체 데이터집합 반복 수행 (Epoch)에서 내부 반복 지표 (Internal Epoch index)  $k_t$ 에 대하여 출력하는 함수  $z(k_t) \in \mathbf{Z}, z(k_t) > 0$ 를 놓고 이를 사용하여 학습률  $\lambda_t$ 를  $\lambda_t = z(k_t)Q_p^{-1}$ 로 놓으면 양자화된 갱신항은 다음과 같다.

$$(\lambda_t h_t)^Q = z(k_t) h_t^Q, \quad h_t^Q \triangleq \left( \frac{1}{Q_p} h_t \right)^Q, \quad h_t^Q \in \mathbf{Q}^n \quad (9)$$

### 3. 양자화 기반 최급 강하법과 적응적 학습률의 기계학습 적용 방법

#### 3.1 양자화 기반 최급 강하법의 단점

식 (9)를 통해 양자화된 갱신항을 정의할 수 있게 되면, 식 (8)에 의하여 양자화된 변수는 모든  $t \in \mathbf{Z}^+$ 에 대하여,  $x_t \in \mathbf{Q}^n$ 을 유지하게 되어 양자화 기반 최급 강하법을 기계 학습에 적용할 수 있다. 그러나, 학습률  $\lambda_t^Q$ 는 다음의 정의에 따라, 매 전체 데이터 집합 반복수행에 대하여 경사 도함수 방향으로 목적함수  $f(x_t^Q + \lambda_t^Q h_t^Q)$ 를 최소화시켜야 한다.

$$\lambda_t^Q = \arg \min_{\lambda^Q} f(x_t^Q + \lambda^Q h_t^Q) \quad (10)$$

그러므로 식 (10)의 과정은 필연적으로 기계학습시, 최적의 학습률을 찾기 위한 내부 루프 (Inner-Loop)를 필요로 하게 되어 단위 전체 데이터 집합 반복수행 때 마다, 내부적으로 반복 수행을 해야 한다. 특히, 양자화를 사용하는 기계학습의 경우, 양자화 오차에서 파생되는 학습 오차가 데이터당 반복 (Iteration) 혹은 전체 데이터 집합 당 반복 (Epoch)에서 더욱 증폭될 수 있으므로, 학습 오차를 최소화시킬 수 있는 최적 학습률을 내부 루프 없이 계산할 수 있는 알고리즘 도출이 필요하다.

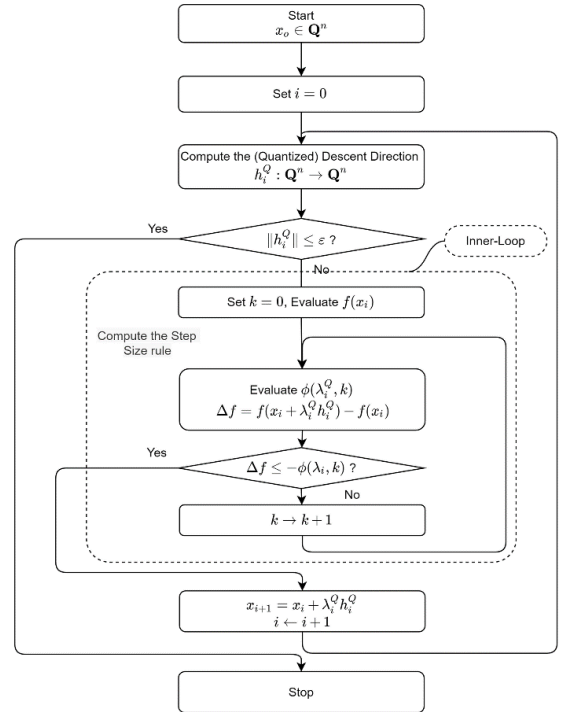


그림 1. 기존 비선형 최적화 기법에 기반한 적응적 학습률을 가진 기계학습의 학습 알고리즘

#### 3.2 적응적 학습률의 기계학습 적용

기존 기계학습에서는 적응적 학습률을 구현하기 위하여 일반적으로 Newton-Rapson 최적화법에 기반한 알고리즘 (AdaDelta, AdaDelta, ADAM 등)이 존재했으나 이들 역시, 근사 Hessian 을 사용하여 경사 도함수의 적용 비율에 대한 변화만을 주었을 뿐, 근본적으로 일정한 학습률을 사용하였다 [6] [7] [8]. 때문에, 현재 제안된 거의 대부분의 확률적 최급

강화법에 의한 학습 방정식으로는 전체 데이터집합 반복 수행에서 국소적으로도 점근적인 수렴성(Asymptotically Convergence)을 보장할 수 없다.

그러므로, 최소한 국소적 점근적 수렴성을 보장하기 위해서는 갱신항을 포함하는 학습 변수에 대한 목적함수  $f(x_t^Q + \lambda_t^Q h_t^Q)$ 와 갱신항을 포함하는 학습변수에 대한 목적함수  $f(x_t^Q)$ 의 차이가 항상 음의 값을 갖도록 다음의 관계가 성립하여야 한다.

$$f(x_t^Q + \lambda_t^Q h_t^Q) - f(x_t^Q) \leq -\phi(\lambda_t^Q, k) \quad (11)$$

식 (11)에서  $\phi(\lambda_t^Q, k) \in \mathbf{R}^+$ 은 내부 루프 반복지시자  $k$ 에 대한 단조감소 함수이다. 따라서, 내부 루프 반복지시자를 전체 데이터 집합 반복 수행 지시자에 통합하여 식 (11)을 만족하면 학습 변수가 갱신되도록 하며, 만족하지 못하면 변수가 갱신되지 않도록 하고, 내부루프 지시자를 증가시켜 함수  $\phi(\lambda_t^Q, k)$ 가 감소하도록 한다. 이를 도식하면 그림 2와 같다.

제안한 알고리즘에서는, 식 (11)을 만족하면 알고리즘은 보통의 기계학습과 같다. 그러나, 식 (11)을 만족하지 못하면, 학습이 이루어지지 않고 더 작은 학습률로 다시 학습하는 것이기 때문에 본 과정은 마치 “Idle and Go”처럼 보인다. 또한 제안한 알고리즘은 만일, 유한한 내부 루프로 학습률을 찾을 수 있다면, 상대적으로 적은 전체 데이터 집합 반복수행으로 학습률을 찾을 수 있으므로 전체 학습 시간에서 차지하는 비중이 크지 않아, 효율적으로 적응적 학습률을 구할 수 있다.

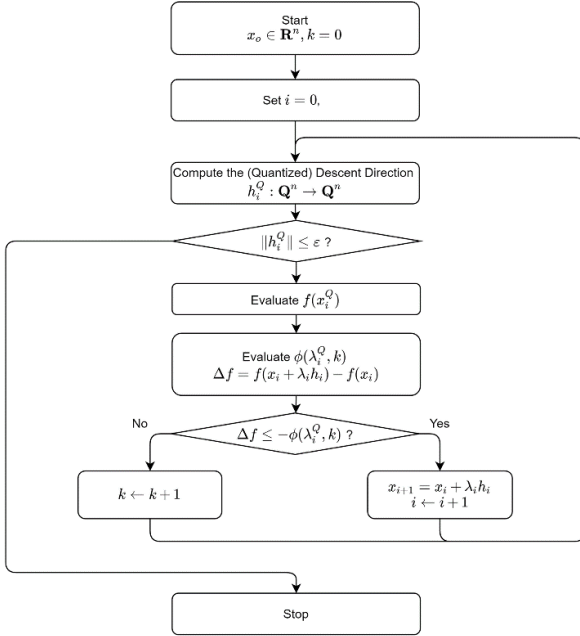


그림 2. 제안한 내부루프 없는 비선형 최적화 기법에 기반한 적응적 학습률을 가진 기계학습의 학습 알고리즘

#### 4. 양자화 오차 보상 방법

일반적으로 탐색 방향 벡터  $h_t \in \mathbf{R}^n$ 에 대한 양자화 벡터  $(h_t)^Q = Q_p h_t^Q \in \mathbf{R}^n$ 가 다음과 같이 구성되었다고 하자.

$$(h_t)^Q = \sum_{i=0}^{n-1} v_i e_i, \quad \forall i \in \mathbf{Z}[0, n-1], v_i \in \mathcal{Q} \quad (12)$$

식 (12)에서  $e_i$ 는 유클리드 공간  $\mathbf{R}^n$ 의 단위 직교 벡터로  $\forall i, j \in \mathbf{Z}[0, n), \|e_i\| = 1, e_i^T e_j = 0 \quad i \neq j$ 이다. 양자화 벡터  $(h_t)^Q$ 의 성분  $v_i$  중 가장 크기가 큰 성분을  $v_m = \max_i \|v_i\|$ 이라 하고 이것의 인덱스를  $m = \operatorname{argmax}_i \|v_i\|$ 라 하자. 이때, 양자화 벡터  $(h_t)^Q$ 에서  $v_m = 0$ 로 놓은 것을  $\bar{v}$ ,  $v_m$ 을 제외한 모든 성분이 0인 벡터를  $\hat{v}$ 라 하고 각각 다음과 같이 정의한다.

$$\bar{v} = \sum_{i=0}^{n-1} (1 - \delta(i-m))v_i e_i, \quad \hat{v} = \sum_{i=0}^{n-1} \delta(i-m)v_i e_i \quad (13)$$

그러므로, 탐색 방향 벡터는 두개의 직교 벡터  $\bar{v}, \hat{v}$ 로 나누어진다.

$$(h_t)^Q = \bar{v} + \hat{v} = \sum_{i=0}^{n-1} (1 - \delta(i-m) + \delta(i-m))v_i e_i \quad (14)$$

기존의 탐색방향 벡터  $(h_t)^Q$ 에서 가장 큰 성분에 대해 직교하는 방향의 벡터를 구하기 위해 다음과 같이 벡터  $z$ 를 놓자

$$z_t = \bar{v} + r \cdot \hat{v}, \quad r \in \mathbf{R} \quad (15)$$

식 (15)에서  $r \in \mathbf{R}$ 는  $\hat{v}$ 에 대한 비례상수로서 이 값을 통해 직교 벡터  $z$ 를 구할 수 있다. 벡터  $z$ 와 벡터  $(h_t)^Q$ 와 직교성을 사용하여  $r$ 은 다음과 같이 구할 수 있다.

$$\begin{aligned} 0 &= \langle (h_t)^Q, z \rangle = \langle \bar{v} + v, \bar{v} + r\hat{v} \rangle \\ &= \bar{v}^2 + (r+1)\langle \hat{v}, \bar{v} \rangle + r\hat{v}^2 \quad \because \langle \hat{v}, \bar{v} \rangle = 0 \\ &= \bar{v}^2 + r\hat{v}^2 \end{aligned} \quad (16)$$

$$\therefore r = -\frac{\bar{v}^2}{\hat{v}^2} = -\frac{\|(h_t)^Q\|^2 - v_m^2}{v_m^2} = 1 - \frac{\|(h_t)^Q\|^2}{v_m^2} \quad (17)$$

그런데,  $|r| < 1$ 이므로 이를 그대로 학습 방정식에 적용하게 되면 정수 값으로 연산이 이루어지지 않으므로 보상 탐색 벡터는 양자화 된 값이 아닌, 일반적인 실수 벡터가 된다. 따라서 비례상수의 양자화를 고려하여 보상 탐색 벡터를 구하여야 한다. 식 (13)에서  $(h_t)^Q = Q_p h_t^Q$ 이므로  $v_m = Q_p v_m^Q$ 를 사용하여 정리하면

$$\hat{v} = \sum_{i=0}^{n-1} \delta(i-m)v_i e_i = Q_p \cdot v_m^Q e_m \quad (18)$$

식 (18)에서  $v_m^Q e_m \triangleq \hat{v}^Q$ 로 놓으면

$$\begin{aligned} z_t &= \bar{v} + r \cdot \hat{v} = \bar{v} + r \cdot Q_p \hat{v}^Q = \bar{v} + Q_p r \cdot \hat{v}^Q \\ &= \bar{v} + Q_p \hat{v}^Q - Q_p \hat{v}^Q + Q_p r \cdot \hat{v}^Q \\ &= (\bar{v} + \hat{v}) + Q_p (r-1)\hat{v}^Q \quad \because \hat{v} = Q_p \hat{v}^Q \end{aligned} \quad (19)$$

식 (13)(17) 그리고  $(h_t)^Q = Q_p h_t^Q$ 이므로,

$$z_t = Q_p h_t^Q - Q_p \frac{\|(h_t)^Q\|^2}{v_m^2} \hat{v}^Q \quad (20)$$

그러므로 식 (9)를 사용하여  $\hat{v}^Q$ 의 계수를 정리하면

$$Q_p \cdot \frac{\|(h_t)^Q\|^2}{v_m^2} = Q_p \cdot \frac{\sum_{i=0}^{n-1} v_i^2}{v_m^2} = \sum_{i=0}^{n-1} Q_p \frac{v_i^2}{v_m^2} \quad (21)$$

그러므로 식 (20)(21)을 사용하여 양자화를 하면

$$(z_t)^Q = Q_p h_t^Q - \left( \sum_{i=0}^{n-1} \left\lfloor \frac{Q_p v_i^2}{v_m^2} \right\rfloor + \left[ \sum_{i=0}^{n-1} \frac{1}{v_m^2} \left( Q_p v_i^2 - v_m^2 \left\lfloor \frac{Q_p v_i^2}{v_m^2} \right\rfloor \right) + 0.5 \right] \right) \hat{v}^Q \quad (22)$$

식 (22)에서  $\left\lfloor \frac{Q_p v_i^2}{v_m^2} \right\rfloor$  는  $\frac{Q_p v_i^2}{v_m^2}$  의 몫이고  $Q_p v_i^2 - v_m^2 \left\lfloor \frac{Q_p v_i^2}{v_m^2} \right\rfloor$  는 나머지이다. 나머지 부분을 다음과 같이 간략화 하면

$$Rem\left(\frac{Q_p v_i^2}{v_m^2}\right) = \frac{1}{v_m^2} \left( Q_p v_i^2 - v_m^2 \left\lfloor \frac{Q_p v_i^2}{v_m^2} \right\rfloor \right) \quad (23)$$

양자화된 직교 보상 탐색벡터는 다음과 같이 쓸 수 있다.

$$(z_t)^Q = Q_p h_t^Q - \left( \sum_{i=0}^{n-1} \left\lfloor \frac{Q_p v_i^2}{v_m^2} \right\rfloor + \left[ Rem\left(\frac{Q_p v_i^2}{v_m^2}\right) \right] \right) \hat{v}^Q \quad (24)$$

식 (24)로 구해진 양자화된 직교 보상 탐색벡터는 양자화 기반 최급 강하법에서  $\phi(\lambda_t^Q, k)$  함수가 충분히 작은 값임에도 학습 변수의 갱신이 이루어지지 않는 경우 탐색 방향벡터  $h_t$  대신 사용하게 된다. 이때, 식 (24)는 양자화 계수  $Q_p$  의 스케일을 가지고 있으므로 이를 기본 양자화 보상 탐색벡터  $z_t^Q \equiv \frac{1}{Q_p} (z_t)^Q$  로 놓고  $h_t^Q$  대신  $z_t^Q$  를 사용하여 탐색한다. 이 때, 탐색결과 새로운 학습 변수로 갱신되면 다시 원래의 최적화 알고리즘을 사용하여 최적화 과정을 지속한다.

## 5. 실험결과

본 알고리즘의 타당성을 실험하기 위하여 대표적인 비선형 최적화 문제인 로젠브록(Rosenbrock) 함수의 최적화 문제에 제안한 양자화 방식을 적용하였다. 실험에 사용한 로젠브록 함수는 다음과 같다[6].

$$\forall x, y \in \mathbf{R}, f(x, y) = (a - x)^2 + b(y - x^2) |_{a=1, b=100} \quad (25)$$

식 (25)의 경우 유리수체로 정의 가능하므로,  $\frac{1}{Q_p} h_t \in \mathbf{Q}$  를 잘 정의(Well-defined)할 수 있다. 최적화 알고리즘은 Quasi-Newton 방법인 BFGS[4]를 사용하여 Python 환경에서 구현하였으며 양자화 계수  $Q_p$  는 표 1 과 같이 낮은 해상도(작은  $Q_p$  값)에서 높은 해상도(큰  $Q_p$  값)까지 수행하였다. 양자화 계수는 2 진수 시스템을 가정하여  $\eta = 1, \rho = 2$  로 놓았다. 실험은 Intel i7-6700K 4GHz CPU 에서 Windows 10 OS 환경에서 Python 3.5 로 프로그래밍 하여 실험하였다.

실험결과 제안한 양자화 알고리즘의 경우, 고정 소수점이 7bit 일때 부터 최적점을 찾아가는 것을 알 수 있었으며 수행시간의 경우 양자화 되지 않은 경우보다 50% 정도 줄어드는 것을 알 수 있다.

Log Qp	Qp	Best Epoch	cost	epsilon	Processing Time
L-BFGS		43	0	0	0.68813205
4	16	1	20.91852	-1.51367	0.00797892
5	32	2	19.42276	0	0.0090034
6	64	2	19.27893	0	0.01196694
7	128	46	0	6.14E-05	0.30318999
8	256	43	0	1.53E-05	0.28124738
9	512	42	0	3.45E-05	0.28921962
10	1024	41	0	3.45E-05	0.27825594

표 1. L-BFGS 알고리즘과 제안한 알고리즘과의 최적화 성능

## 6. 결론

본 논문에서는 임베디드 시스템에서 비선형 최적화 방법론을 기반으로 양자화 학습을 최적하게 수행하기 위해 내부 루프 없는 적응적 학습률을 사용하는 학습 알고리즘과 양자화 오차를 최소화시킬 수 있는 보상 탐색 방법론을 제안하였다. 실험 결과 제안한 방법은 양자화 오차를 효과적으로 줄일 수 있음을 비선형 최적화 문제를 통해 확인할 수 있었다. 향후 과제로는 제안한 알고리즘의 수학적 분석을 통한 수렴성 및 특성 분석과 상대적으로 대규모 데이터 및 대규모 기계학습 구조에서, 효과적으로 적용됨을 보이는 것이다.

## 7. 감사의 글

본 논문은 2019 년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No.2017-0-00142, 스마트기기를 위한 온디바이스 지능형 정보처리 가속화 SW 플랫폼 기술 개발)

## 참고문헌

- [1] R. M. Gray, and D. L. Neuhoff, "Quantization.", IEEE Tr. Inform. Theory, 44 (6):2325-2383, 2006.
- [2] D. Alistarh, D. Grubic, L. J. Ryota, and V. Milan, "QSGD: Communication-efficient sgd via gradient quantization and encoding.", Advances in Neural Information Processing Systems, 30:1709-1720, 2017
- [3] G. Tenenbaum, 'Introduction to Analytic and Probabilistic Number Theory', Academic mathematical Society, 2014.
- [4] D. G. Luenberger, Y. Ye, 'Linear and Nonlinear Programming', Springer, 2015.
- [5] S. Sra, S. Nowozin, S.J.Wright, 'Optimization for Machine Learning', MIT press, 2012.
- [6] J. Duchi, E. Hazan, and Y. Singer. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. The Journal of Machine Learning Research, 2011.
- [7] M. Zeiler, "ADADELTA: an adaptive learning rate", arXiv preprint, <https://arxiv.org/abs/1212.5701>, arXiv:1212.5701, 2012.
- [8] D. Kingma, J. Ba. Adam: A Method for Stochastic Optimization. International Conference for Learning Representations, 2015.
- [9] S. M. Goldfeld, R. E. Quandt, and H. F. Trotter, "Maximization by Quadratic Hill-Climbing", Econometrica, pp. 541-551, Jul., 1966.