

# 개선된 에너지 함수를 가지는 신경망 기반의 라우팅 알고리즘

정회원 박 동 철\*, 금 교 린\*\*

## A Neural Network-based Routing Algorithm With an Improved Energy Function

Dong-Chul Park\*, Kyo-Reen Keum\*\* *Regular Members*

### 요 약

효율적인 통신망에서의 라우팅을 위해, 개선된 에너지 함수를 가지는 Hopfield 신경망에 의한 알고리즘이 본 논문에서 제안되었는데, 보다 높은 최적경로 형성과 안정적 수렴의 결과를 목적으로 한다. 20-50 개의 노드를 가지며, 무작위 연결 비용이 인가되는 3,000 개의 통신망에 대한 실험의 결과에서 볼 때, 기존의 신경망을 이용한 알고리즘들이 노드 수가 많은 망 환경에서 수렴하지 않거나, 최적경로가 형성이 되지 않은 경우가 많았지만 제안된 알고리즘은 최적경로 형성에서 기존 알고리즘 보다 약 65%의 개선을 하였고 또한 기존 알고리즘 보다 약 50% 정도 수렴이 잘 되는 것이 확인되었다.

Key Words : Hopfield Neural Network, Energy function, Optimality, Convergence rate

### ABSTRACT

A routing algorithm using the Hopfield Neural Netork (HNN) is proposed in this paper. The proposed algorithm modifies the energy function for achieving the optimality of the solution and higher convergence rate. Experimental results show that the proposed algorithm outperforms convensional methods both in optimality and convergence.

### I. 서 론

최근의 다양한 통신망 환경의 특징은 하나의 출발지로부터 여러 도착지가 존재하는 경우가 많고, 또한 망의 규모가 크다<sup>[1]</sup>. 이러한 환경에서 망관리 시스템의 성능을 결정짓는 중요한 요소들 중의 하나인 최적 경로 라우팅을 위해 신경망은 좋은 도구로 인식되는데 이는 신경망이 하드웨어로 구현되었을 때, 뉴런들이 빠른 연산 속도로 인한 실시간 응용등의 장점을 가지고 있기 때문이다. 신경망을 이용하여 최단 경로를 찾는 라우팅 알고리즘을 처음

소개한 것은 Rauch와 Winarske였다<sup>[2]</sup>. Zhang과 Thomopoulos는 후에 주어진 망에서 전체 노드를 경유하여 최단 경로를 찾는 알고리즘을 제안하였다<sup>[3]</sup>. 한편, Ali와 Kamoun(AR)은 망구조의 정보에 적응적인 알고리즘을 소개하였다<sup>[4]</sup>. 루프나 파티션을 제거하는 라우팅 방식이 Park과 Choi(PC) 알고리즘에서 소개되어, 규모가 비교적 큰 망에서의 최적 경로 형성에 제한적이지만 기존의 방법들에 비해 성능의 개선을 보였다<sup>[5]</sup>. 이후 빠른 수렴과 최단 경로 형성을 위해 Ahn과 Ramakrishna(AR)가 제안한 알고리즘이 소개 되어, 수렴의 속도를 증가시키고, 최

\*, \*\* 명지대학교 정보공학과 지능컴퓨팅 연구실  
 논문번호: KICS2004-07-097, 접수일자: 2004년 7월 12일

적 경로의 확률을 증가시키는 것으로 알려져 왔다 [6].

본 논문에서는 기존의 알고리즘들에서 다루어진 망에서 보다 규모가 큰 망에서 수렴성을 증가시키며, 최단 경로를 찾기 위해 새로운 Lyapunov Energy 함수를 제안하는데, 본 논문의 2장에서는 홉필드 신경망을 이용한 라우팅 문제의 방법과 개요에 대해서 살펴본다. 3장에서는 최적 라우팅에 관한 기존의 알고리즘들에 대한 비교가 주어지며, 4장에서는 제안 알고리즘을 설명하고, 5장에서는 실험을 통해 제안된 알고리즘의 성능을 기존의 라우팅 알고리즘과 비교하며, 6장에서는 결론이 주어진다.

## II. 라우팅 문제 정의

통신망에서의 연결 비용 행렬은  $L=[L_{ij}]$ 로 표현될 수 있는데, 여기서  $L_{ij}$ 는  $i$ 노드에서  $j$ 노드까지의 링크비용으로,  $i$ 노드와  $j$ 노드 사이의 지연시간, 대역폭, 그리고 교통부하등 통신의 비용을 객관적으로 평가할 수 있는 변수들을 종합적으로 포함한다. 라우팅 문제를 일반적 조합 최적화 문제로 공식화 하기위해 모든 대각선 행렬의 값이 0을 갖는 ( $n \times n$ ) 행렬이 필요하다. 여기서  $n$ 값은 망이 가지고 있는 노드의 수이다. 행렬이 가지고 있는 값은 연결상태를 나타내고, 두개의 인자 ( $i, j$ )로 표시한다. 연결상태  $V_{ij}$ 는 노드  $i$ 와 노드  $j$ 사이엔 연결이 있으면 "1"이고 없으면 "0"으로 정의 한다. 이를 이용하여 최단 경로 라우팅에 대한 문제는 다음과 같은 제한 조합 최적화 문제로 표현된다<sup>[2]</sup>.

$$\text{Minimize } \sum_{i=1}^N \sum_{j=1}^N L_{ij} V_{ij} \text{ for } (s \rightarrow d) \quad (1)$$

$$\left( \sum_{j=1}^N V_{ij} - \sum_{k=1}^N V_{ki} \right) = \begin{cases} 1, & \text{if } i = s \\ -1, & \text{if } i = d \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

여기서,  $s$ 는 출발 노드이고  $d$ 는 도착 노드로 정의된다.

## III. 홉필드 신경망을 이용한 라우팅 알고리즘

제한 조합 최적화 문제를 신경망으로 접근하는 방법은 Hopfield와 Tank<sup>[7]</sup>에 의해서 처음 소개되었고, 그 후에 주어진 출발지와 목적지 사이에 최단

경로를 신경망으로 찾는 방법이 Rauch와 Winarske에 의해 소개 되었다<sup>[2]</sup>. 그러나 이 알고리즘은 최단 경로를 구성하는 노드의 수를 알 수 가 없고, 노드의 수를 미리 결정하여 최단 경로를 찾기 때문에 최적의 해를 찾을 수 있는 경우가 매우 적다. 이 알고리즘의 문제점을 해결하기 위해 Zang과 Thomopoulos는 모든 경로를 경유하여 최단 경로를 찾는 방법을 제안하였다<sup>[3]</sup>. 이 알고리즘은 에너지가 최소가 되는 방향으로 뉴런을 학습시켜서 최단 경로를 찾았다. 그러나 이 알고리즘의 단점은 링크 비용이 변하면 이에 따른 연결 저항의 값을 상황에 맞게 변경해줘야 하기 때문에 시변 망에 알맞지 않다. 또한, 모든 노드를 경유해야 최단 경로를 찾는다는 문제점이 있다.

최단 경로에 필요한 노드 수를 알 수 없는 것과 모든 노드를 경유해야 하는 기존의 알고리즘의 단점을 보완한 방법이 AK에 의해 제안되었다<sup>[4]</sup>. 즉, 최단 경로를 찾는 실질적인 방법이 제시된 것이다. 이 알고리즘에서 제안한 에너지 함수는 식(3)과 같이 표현된다.

$$\begin{aligned} E_{AK} = & \frac{\mu_1}{2} \sum_{i=1}^N \sum_{j=1}^N L_{ij} \cdot V_{ij} + \frac{\mu_2}{2} \sum_{i=1}^N \sum_{j=1}^N \gamma_{ij} \cdot V_{ij} \\ & + \frac{\mu_3}{2} \sum_{i=1}^N \left( \sum_{j=1}^N V_{ij} - \sum_{j=1}^N V_{ji} \right)^2 \\ & + \frac{\mu_4}{2} \sum_{i=1}^N \sum_{j=1}^N V_{ij} (1 - V_{ij}) \\ & + \frac{\mu_5}{2} (1 - V_{ds}) \end{aligned} \quad (3)$$

여기서  $\mu_1$ 항은 링크가 존재하는 곳의 총 비용을 최소화 하도록 하고,  $\mu_2$ 항은 링크가 존재 하지 않는 곳은 경로가 형성되지 않도록 하고,  $\mu_3$ 항은 동일한 노드를 다시 경유하는 것을 방지하며,  $\mu_4$ 항은 수렴을 빨리 할수 있게 해준다. 마지막으로  $\mu_5$ 항은 출발지와 도착지사이의 링크를 경로에 포함시키도록 한다. 한편,  $\gamma_{ij}$ 는 노드  $i$ 와 노드  $j$ 사이엔 망 구조상의 연결이 있으면 "0", 없으면 "1"로 망 구조상의 연결상태를 나타낸다.

하지만 AK의 알고리즘 또한 최단경로 설정의 문제에 있어서 배제되어야 하는 출발지 도착지간의 고립된 경로인 루프나 파티션이 생기는 경우가 있고, 규모가 큰 망 환경에서 수렴이 안되는 경우가

많다. AK 알고리즘의 단점을 보완하여 경로 설정 상에 루프나 파티션의 존재 가능성을 제거하는 방법이 PC에 의해 제안되었다<sup>5)</sup>. 이 알고리즘이 가지고 있는 에너지 함수는 식(4)과 같이 표현된다.

$$\begin{aligned}
 E_{PC} = & \frac{A}{2} \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N L_{ij} \cdot V_{ij} + \frac{B}{2} \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N \gamma_{ij} \cdot V_{ij} \\
 & + \frac{C}{2} \sum_{i=1}^N \left( \sum_{\substack{j=1 \\ j \neq i}}^N V_{ij} - \sum_{\substack{j=1 \\ j \neq i}}^N V_{ji} - \phi_i \right) \\
 & + \frac{D}{2} \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N V_{ij} \cdot (1 - V_{ij}) \\
 & + \frac{F}{2} \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N V_{ij} \cdot V_{ji}
 \end{aligned} \quad (4)$$

여기서 A, B, C와 D항은 AK의 알고리즘에서의  $\mu_1, \mu_4$ 와 같은 의미를 갖는다. 한편 F항은 하나의 노드에서 경로가 나가도록 설정되었을 때 그 노드로 경로가 다시 형성되지 못하도록 하는 역할을 한다.

최근에 AR은 경로 형성의 수렴속도를 크게 향상시킨 다음의 에너지 함수를 제안하였다<sup>6)</sup>.

$$\begin{aligned}
 E_{AR} = E_{PC} + & \frac{\mu_6}{2} \left( \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N \left( \sum_{\substack{k=1 \\ k \neq i, j}}^N V_{ik} - 1 \right) \cdot V_{ij}^2 \right) \\
 & + \frac{\mu_7}{2} \left( \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N \left( \sum_{\substack{k=1 \\ k \neq i, j}}^N V_{kj} - 1 \right) \cdot V_{ij}^2 \right)
 \end{aligned} \quad (5)$$

여기서 보듯이  $E_{AR}$ 은  $E_{PC}$ 에  $\mu_6$ 과  $\mu_7$ 항을 추가하였는데, 그 의미는 하나의 노드에서 다른 노드로 경로가 형성될 가능성이 생기는 경우 뉴런의 진화를 강제적으로 빨리 시켜서 수렴의 속도를 향상시켰다. 하지만, 이 알고리즘은 하나의 노드에 링크가 존재하는 다른 노드 간에 경로설정 가능성이 하나의 노드에만 존재 할 경우에는 수렴속도가 빠르게 이루어지는 반면 두개 이상의 노드로 경로설정 가능성이 생기면 루프나 파티션이 발생되거나 전혀 수렴되지 않는다.

#### IV. 수정된 에너지 수식을 가진 홉필드 신경망을 이용한 라우팅 알고리즘

본 논문에서는 홉필드 신경망을 이용하여 좀더 개선된 라우팅 알고리즘을 얻기위해 에너지 함수를 수정하였는데, 식(6)과 같이 표현된다.

$$\begin{aligned}
 E = & \frac{A}{2} \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N \left( V_{ij} \cdot \sum_{\substack{k=1 \\ k \neq i}}^N (1 - L_{ik}) \cdot L_{ij} \right) \\
 & + \frac{B}{2} \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N \gamma_{ij} \cdot V_{ij} \\
 & + \frac{C}{2} \sum_{i=1}^N \left( \sum_{\substack{j=1 \\ j \neq i}}^N V_{ij} - \sum_{\substack{j=1 \\ j \neq i}}^N V_{ji} - \phi_i \right) \\
 & + \frac{D}{2} \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N V_{ij} \cdot (1 - V_{ij}) \\
 & + \frac{F}{2} \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N \left( V_{ij} \cdot \sum_{\substack{k=1 \\ k \neq i}}^N V_{ki} \right)
 \end{aligned} \quad (6)$$

식(6)에서  $\phi_i$ 는 식(7)로 정의된다.

$$\phi_i = \begin{cases} 1, & \text{if } i = s \\ -1, & \text{if } i = d \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

식(6)의 수정된 에너지 함수에서 A항은 식(1)의 비용에 관한 목적함수를 최소화하는 것으로, 식(4)의  $E_{pc}$ 를 수정하여 링크가 존재하는 곳의 총 비용을 최소화 하도록 하며, 주변 노드에 연결된 링크 비용을 모두 고려하여 가장 적은 비용을 가진 링크 비용으로 경로가 형성되도록 한다. 한편, B, C, D항은 식(4)의 경우와 같다. 한편, F항의 수정된 부분은 노드 i에서 노드 j로 경로가 형성되었을때 노드 j에서 노드 i로 경로 형성을 방지하고 또한, 노드 j를 제외한 노드 i와 링크가 존재하는 모든 다른 노드에 대해서도 노드 i로의 경로 형성을 방지한다.

#### V. 실험 및 결과

본 논문에서는 홉필드 신경망을 이용한 라우팅 알고리즘의 성능을 기존의 세 가지 알고리즘 (AK가 제안한 방법, PC가 제안한 방법, AR이 제안한 방법)과의 성능을 비교한다. 다양한 망 구조를 발생 시켜서 Dijkstra 알고리즘<sup>8)</sup>과 비교하여 최단 경로에 대한 비교 알고리즘들의 최적성을 분석하고, 각 알고리즘의 큰 망 환경에서의 수렴 비율도 비교한다. 여기서 Dijkstra 알고리즘<sup>8)</sup>의 결과는 주어진 망에서의 최적 경로로 가정되는데, 이는 일반적으로 통용되는 방법이다<sup>6)</sup>.

에너지 함수의 계수는 최단 경로 형성에 큰 영향을 주기 때문에 적절한 값을 설정해 주어야 한다. 비록 뉴런의 상태가 지역 최소점에 빠질 수 있지만,

에너지 함수의 적절한 계수 설정으로 이런 점을 막을 수 있다. 여기서 사용된 계수값을 결정한 방법은 기존의 연구결과에서 보듯이 특별하게 이론적으로 규정된 방법은 없다. 다만 실험을 통해 최적의 계수를 구하는 방법이 존재할 뿐이지만, 대체로 이제까지 알려진 방법에 의하면 A는 비교적 큰 값으로 설정되지만, 링크가 존재하지 않는 노드간의 경로 설정이 되지 않도록 하기위해 B보다는 적은 값으로 설정한다. C는 출발 노드에서 시작한 경로가 도착 노드로 지속적인 연결을 할 수 있도록 하는 역할이므로 A보다는 큰 값으로 하지만, B보다는 작은 값으로 설정해야 한다. D와 F는 그 값이 클 경우 경로 설정 시 준 최적 경로가 될 확률이 크기 때문에 A값 보다는 아주 작게 설정한다. 본 실험에 사용된 계수는 표 1과 같다.

표 1. 실험에서 사용된 계수값

Parameter	A	B	C	D	F	$\tau$	$\Delta t$	$\lambda$
Value	950	2500	1900	100	500	1	$10^{-4}$	1

실험에서는 노드의 수가 12~25개, 26개~40개 그리고 41개~50개를 가지는 망을 각각 1,000개씩 생성하여 총 3,000 개의 망에 대한 실험을 수행하였다. 각 망의 링크 비용은 무작위로 부여하여 실험결과의 일반성을 최대한으로 유지 시켰다.

표 2. 수렴비율(%) 비교

algorithm \ 노드수	그룹 1 (12~25)	그룹 2 (26~40)	그룹 3 (41~50)
AK[4]	99.28	98.80	42.00
PC[5]	99.57	99.20	46.13
AR[6]	98.85	76.20	13.34
Proposed	99.85	99.40	94.23

표 2의 수렴 비율은 각 알고리즘을 안정 상태에 도달할 때까지 반복 계산을 충분히 한후,(20,000번) 주어진 출발지와 도착지간의 경로 형성이 이루어지는 비율을 말한다. 표 2에서 보듯이 본 논문에서 제안한 알고리즘은 망이 커져도 수렴 비율이 많이 떨어지지 않았다. 하지만 AK, PC 알고리즘은 큰 망에서 수렴 비율이 떨어졌고, 특히 AR이 제한한 알고리즘은 망이 41개 이상의 노드를 가지고 있는 망 구조에서 수렴 비율이 매우 낮았다.

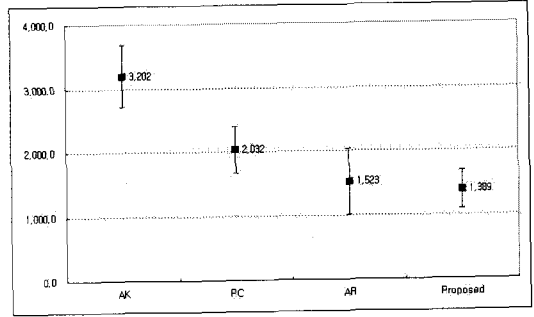


그림 1. 각 알고리즘의 계산량 비교

그림 1은 각 알고리즘이 출발지와 도착지간의 경로 형성이 된 후, 수렴에 성공한 결과만 가지고 각 노드의 계산량인 iteration의 수를 비교하였다. 그래프에서 중간점은 iteration의 평균값이고 위아래의 선은 표준편차이다. 그림 1에서 보듯이 제안된 알고리즘의 평균 계산량이 가장 적었다. AR 알고리즘의 경우 계산량이 다른 알고리즘에 비교해서 가장 적은 경우가 간혹 있었지만 편차가 크게 나타났고 표 2의 결과에서처럼 수렴되지 못한 결과는 제외하여 분석하였으므로 계산량이 항상 적은 것은 아니다.

표 3. 결과의 최적성(%) 비교

algorithm \ 노드수	그룹 1 (12~25)	그룹 2 (26~40)	그룹 3 (41~50)
AK[4]	82.61	46.13	15.00
PC[5]	82.65	63.76	24.20
AR[6]	65.81	52.63	11.21
Proposed	87.97	82.20	79.99

표 3은 각 알고리즘에서 수렴된 결과에 대해 총 연결비용을 나타낸 것인데, 각 경우에 대해 Dijkstra<sup>[8]</sup>의 결과에 의한 총 연결비용을 100%라고 가정했을때의 상대적인 총 연결비용의 역수이다. 따라서 상대 연결비용 비율의 값이 높을수록 최적 결과에 가까운 결과를 나타낸다. 표 3에서 보듯이 제안한 알고리즘의 최단 경로 형성에 의한 최적성은 망이 커져도 크게 차이가 없었다.

기존의 AK와 PC 알고리즘은 망이 커지는 상황에서 수렴 비율과 최적성이 떨어졌다. 이는 망 구조에 따라 굉장히 민감하게 반응하여 불안정한 것을 의미한다. 특히, AR이 제한한 알고리즘은 큰 망 환

경에서는 수렴이 현저하게 저하되었고, 경로형성의 최적성도 다른 알고리즘 보다 많이 저하되었다. 이러한 결과는 [6]에서 보여진 것과 많은 차이를 보이는데, 이는 [6]의 결과가 14개 노드를 가지는 망에 대한 실험의 결과이기 때문으로 추정된다. 즉, AR의 방법은 노드수가 많은 망에서는 최적성에 문제점을 보인다.

위의 표 2, 3은 노드들간의 링크비용을 무작위로 정한 3,000 개의 망에 대한 실험의 결과이므로, 결과에 대한 공정성과 신빙성을 담보한다고 할 수 있다. 본 논문에서 제안된 알고리즘은 뉴런간의 모든 정보를 고려하여 뉴런을 진화시키므로 망 구조가 변하거나 커지는 상황에서도 안정적인 수렴을 하여 기존 알고리즘보다 2배나 높은 수렴 비율을 보인다. 또한 큰 망에서 제안된 알고리즘은 AR의 알고리즘보다 약 8배의 경로 형성의 최적성을 보였다.

## VI. 결론

홉필드 신경망에서의 에너지함수를 새로이 제안하여, 규모가 큰 망에서도 수렴성과 경로 최적성을 높일 수 있는 알고리즘이 본 논문에서 제안되었다. 새로이 제안된 에너지 함수는 연결된 노드간의 뉴런을 진화시킬때, 하나의 노드에 대한 정보만이 아닌 주변에 링크가 존재하는 모든 노드에 대한 정보를 이용한 점이 기존의 알고리즘들과의 차별성이 있다고 할 수 있다. 따라서, 본 논문에서 제안된 알고리즘의 특징은 기존의 알고리즘들이 가진 규모 큰 망환경에서 최적경로를 찾는 어려움이 있는 것을 획기적으로 보완하였다. 따라서, 본 논문에서 제안된 최적경로 탐색 알고리즘은 현재 컴퓨터 통신망이나 통신 단말기의 이동으로 인한 크고 변화가 많은 망에서 실시간 경로 설정에 매우 유용할 것으로 사료된다.

## 참고 문헌

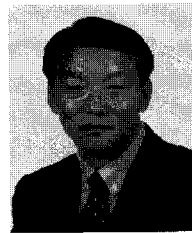
- [1] X. Lin and L. Ni, "Multicast Communication in Microcomputer Networks," *IEEE Tr. Para. and Dist. Sys.*, Vol. 4, No. 10, pp.1105-1117, 1993.
- [2] H. Rauch and T. Winarske, "Neural Networks for Routing Communication Traffic," *IEEE Cont. Syst. Mag.*, pp.26-30, April 1988.
- [3] L. Zhang and S. Thomopoulos, "Neural

Network Implementation of the Shortest Path Algorithm for Traffic Routing in Communication Networks," *Proc. IJCNN* pp.-591, 1989.

- [4] M. Ali and F. Kamoun, "Neural Networks for Shortest Path Computation and Routing in Computer Networks," *IEEE Tr. Neural Networks*, Vol. 4, No.6, pp.941-953, 1993.
- [5] D. Park and S. Choi, "A Neural Network based Multi-destination Routing Algorithm for Communication Network," *Proc. INCNN* pp.1673-1678, 1998.
- [6] C. Ahn and R. Ramakrishna, "Neural Network Based Near-Optimal Routing Algorithm," *Proc. ICONIP*, Vol. 4, pp.1771-1776. 2002.
- [7] J. Hopfield and D. Tank, "Neural Computations of Decisions on Optimization Problem," *Bio. Cyber*, Vol. 52, pp.141-152, 1986.
- [8] W. Stallng, *High-Speed Networks: TCP/IP and ATM Design Principles*, Prentice Hall, Inc., 1998.

박 동 철(Dong-Chul Park)

정회원



1980년 2월 서강대학교 전자공학과(공학사)

1982년 2월 한국과학기술원 전기 및 전자공학과(공학석사)

1990년 6월 Univ. of Wahsington (Seattle), Elec-

trical Engineering (Ph.D.)

1990년 8월~1994년 2월 조교수, Florida Int'l Univ. Dept. of Eelct. and Comp. Eng.

1994년 3월~현재 명지대학교 정보공학과 교수

1997년~2000년 *IEEE Tr. on Neural Networks*, Associate Editor

1999년~현재 *IEEE Senior Member*

<관심분야> 신경망 알고리즘 개발, 음성인식, 신경망의 금융공학에의 응용

금 교 린(Kyo-Reen Keum)

정회원



2002년 2월 명지대학교 제어  
계측공학과(학사)

2004년 2월 명지대학교 정보  
제어공학과(석사)

2004년 2월~현재 (주)SDF  
technology

<관심분야> 신경망, 패턴인

식, UNIX system