

SLAM

(Simultaneous Localization and Mapbuling)

EKF(Extended Kalman Filter)

■ 예측단계 Prediction

로봇에 입력된 속도 명령을 이용하여 로봇의 현재 위치를 추정하는 단계

■ 관측단계 Measurement

예측된 로봇의 현재 위치에서 관측 할 수 있는 특징점의 위치를 계산하고
이를 센서를 통해 실제로 측정한 특징점의 위치와 비교하여
그 오차만큼 로봇의 현재위치와 특징점의 위치를 수정하는 단계

■ 단점

로봇 시스템에서 발생하는 노이즈와 로봇에 사용한 센서에서 발생하는 노이즈에 대한 정확한 모델이 필요,
노이즈 모델에 따라 성능 크게 차이 남

실제 로봇 구동 시에는 가우시안 백색 잡음 뿐만 아니라
다른 형태의 잡음이 로봇 시스템에 영향을 줄 수 있어 성능 크게 저하

특징점의 개수의 제곱에 비례하여 계산량이 늘어나기
때문에 특징점의 개수가 많은 공간에서 처리시간이 길어지는 단점

해결하고자 하는 과제

신경망 - 로봇의 궤적을 계산

진화연산 - 로봇의 자세변화에 따라 발생하는 오차를 보정하여 정확한 지도 작성이 가능

■ 효과

로봇과 센서의 잡음 모델에 따라 신경망의 연결 강도를 최적으로 조절하여 지도를 작성하기 때문에
잡음 모델이 가우시안 기반의 백색잡음이 아니더라도 정확한 지도를 작성할 수 있음

예측한 특징점과 실제 관측된 특징점이 서로 같은 특징점인지 확인 절차 없이도 정확한 지도를 얻을 수 있는 효과

신경망과 진화연산을 이용한 로봇의 환경 지도 작성 방법.

[a] **신경망을 이용하여** 상기 **로봇의 궤적**을 계산하는 단계;

[a1] 상기 로봇에 장착된 **인코더 센서**로부터 계산된

상기 **로봇의 현재 위치 정보**와 **바로 이전 시점의 위치 정보**의 차를 이용하여 얻은 **로봇의 자세 변화값**;

현재 시점에서의 특징점의 위치와 **바로 이전 시점에서의 특징점의 위치**를 상기 신경망에 **입력**하는 단계;

[a2] 상기 신경망에서 **로봇의 자세 변화에 대한 오차**를 출력하고

상기 로봇의 자세 변화에 대한 **오차**를 이용하여 **현재 위치**

정보와 이전 위치 정보의 차를 통해 상기 로봇의 자세 변화 값을 **수정**하는 단계;

[a3] 상기 로봇의 이전 위치에 상기 수정된

자세 변화 값을 더하여 **오차가 보정된 현재 위치**를 계산하는 단계;

[a4] **오차가 보정된 로봇의 궤적**을 계산하는 단계;

[b] 상기 계산된 궤적을 바탕으로 상기로봇이 관측한 **특징점들의 지도**를 작성하는 단계;

하나의 특징점을 여러 위치에서 관측 →

각 특징점들의 위치가 중복되어 표시될 수 있고 어느 정도 흩어져서 분포

센서 시스템의 잡음에 의해 관측된 특징점의 그 위치가 어느 정도 흩어져서 측정됨

로봇 시스템의 잡음이 더해져 주행하면 잡음 모델이 더해져서 결정

[c] 상기 작성된 지도를 **평가**하여 **지도 작성을 종료** 할 것인지 **판단**하는 단계;

[b] 단계에서 만들어진 특징점 지도에서 특징점들을 그

위치에 따라 서로 가까운 곳에 위치한 특징점들끼리 **Clustering 클러스터링**

각 특징점의 **N개 클러스터 (cluster)** 들의 분산 및 각 클러스터들의 분산의 총합으로 계산

$$\sigma_i^x = \sum_{j=1}^{M_i} \frac{(\mu_i^x - f_{i,j}^x)^2}{M_i}$$

$$\sigma_i^y = \sum_{j=1}^{M_i} \frac{(\mu_i^y - f_{i,j}^y)^2}{M_i}$$

$$\text{Cost} = \sum_{i=1}^N (\sigma_i^x + \sigma_i^y)$$

각 **N개 클러스터**들의 분산의 총합을 소정의 기준값과 비교하여

지도작성을 종료할 것 인지 보다 정확한 로봇의 궤적을 계산할 것인지를 판단

[d] 상기 작성된 지도를 평가한 결과 소정의 기준을 만족하지 않는 경우

진화연산 기법을 사용하여 상기 **신경망** 내부의 노드들의 **연결 상태를 조절**하는 단계;

진화연산 (Evolutionary Computation)

이 기법은 유리한 형질을 가진 개체들이 더 많이

생존하게 되고 이들 개체들이 자손을 낳아 개체의 수를 늘려가는 과정에서

더 좋은 형질로의 진화가 이루어지는 유전 법칙에서 영감을

얻어 개발된 기술로서 조합 최적화 문제를 푸는데 이용된다.

신경망을 진화 연산에 적합한 염색체의 형태로 변환시킴

신경망은 7 개의 입력 노드로 구성된 **입력층**

다수의 은닉 노드로 구성된 **은닉층**

3 개의 출력 노드로 구성된 **출력층**

입력층을 구성하는 7 개의 로봇의 위치 $[dx, dy, d\theta]$,
특정점의 위치 $[f_x^t, f_y^t]$ 입력 노드는 식과 같이 벡터 형태로 표시

$$\mathbf{x}_{in} = [x_1, x_2, x_3, x_4, x_5, x_6, x_7]^T = [dx, dy, d\theta, f_x^t, f_y^t, f_x^{t-1}, f_y^{t-1}]^T$$

이렇게 구성된 입력 벡터는 입력층과 은닉층을 잇는
연결에 대한 연결 강도 값이 곱해져서 은닉층에 대한 입력 벡터 $\mathbf{u}$ 를 구성한다
입력 노드의 수가 n 개, 은닉 노드의 수가 m 개인 경우 식을 이용하여
신경망의 **은닉층에 대한 입력 벡터 $\mathbf{u}$** 를 계산할 수 있다

$$\mathbf{u} = \mathbf{W}^h \cdot \mathbf{x}_{in}$$

$$= \begin{pmatrix} w_{1,1}^h & w_{1,2}^h & \cdots & w_{1,n}^h \\ w_{2,1}^h & w_{2,2}^h & \cdots & w_{2,n}^h \\ \vdots & \vdots & \ddots & \vdots \\ w_{m,1}^h & w_{m,2}^h & \cdots & w_{m,n}^h \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

n : 입력층 개수

m : 은닉층 개수

$\mathbf{W}^h$ 는 신경망의 입력층과 은닉층을 잇는 연결에 대한 연결 강도 값들로 이루어진 행렬

은닉층에 대한 출력 벡터 $\mathbf{h}$ 는 위 식을 통해 계산된 은닉층에 대한 입력 벡터 $\mathbf{u}$ 를
실행함수 (activation function) σ 에 입력했을 때의 출력으로 계산된다.

$$\mathbf{h} = \sigma(\mathbf{u}) = \frac{1 - e^{-a\mathbf{u}}}{1 + e^{-a\mathbf{u}}}$$

은닉층 출력 벡터 $\mathbf{h}$ 는 은닉층과 출력층을 잇는 연결에 대한 연결 강도가 아래식과 같이 곱해져서
출력층 입력 벡터 $\mathbf{o}$ 가 계산된다.

$$\mathbf{o} = \mathbf{W}^o \cdot \mathbf{h}$$

$$= \begin{pmatrix} w_{1,1}^o & w_{1,2}^o & \cdots & w_{1,s}^o \\ w_{2,1}^o & w_{2,2}^o & \cdots & w_{2,s}^o \\ \vdots & \vdots & \ddots & \vdots \\ w_{r,1}^o & w_{r,2}^o & \cdots & w_{r,s}^o \end{pmatrix} \cdot \begin{pmatrix} h_1 \\ h_2 \\ \vdots \\ h_s \end{pmatrix}$$

s : 은닉층 개수

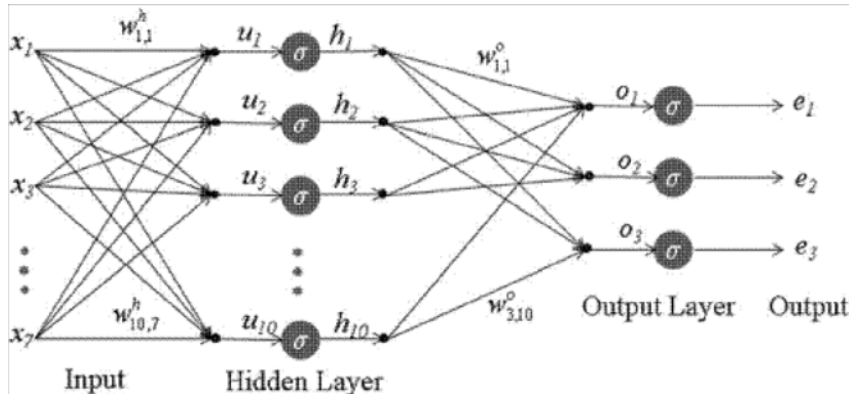
r : 출력층 개수

출력층 입력 벡터 $\mathbf{o}$ 를 출력층의 실행 함수 σ 에 입력하면

최종적으로 로봇이 $x(t-1)$ 지점에서 $x(t)$ 지점으로 이동했을 때 발생한
로봇의 위치 오차를 나타내는 **신경망 출력 벡터 $\mathbf{e}$** 가 계산된다.

$$\mathbf{e} = \sigma(\mathbf{o})$$

신경망은 입력 벡터 x 가 주어졌을 때 출력 벡터 e 를 계산하고,
출력 벡터 e 는 로봇의 위치 오차를 나타내는 값으로서
로봇이 $x(t-1)$ 에서 $x(t)$ 로 이동할 때 발생하는 오차를 보정하는 역할을 한다



- [d1] 상기 신경망의 노드를 잇는 연결 강도 값을 하나의 배열로 만든 **염색체 (chromosome)**를 N 개 생성하고 이를 **초기화**하는 단계;
- [d2] 상기 N 개의 염색체로부터 **돌연변이**를 통해 자손을 생성하여 **$2N$ 개의 염색체**를 생성하는 단계;
- [d3] 상기 $2N$ 개의 염색체를 상기 신경망에 적용하여 상기 로봇의 궤적을 계산하고 **특징점에 대한 클러스터들의 분산의 총합**을 계산하여 **평가**하는 단계;
- [d4] 상기 계산된 **분산의 총합의 결과**에 따라 상기 $2N$ 개의 염색체 중 **새로운 N 개의 염색체**를 선택하는 단계;
상기 $2N$ 개의 염색체 중 상기 클러스터들의 분산의 총합이 작은 N 개의 염색체를 선택하는 단계;
- [d5] 상기 선택된 N 개의 염색체 중 상기 클러스터들의 분산의 총합이 가장 작은 값이 소정의 **정지조건**을 만족하는지 여부를 **판단**하는 단계;
정지조건을 만족하는 경우 상기 클러스터들의 분산의 총합이 가장 작은 염색체를 상기 신경망에 적용하여 얻은 로봇의 궤적과 특징점을 최종적인 로봇의 지도로서 **출력**하는 단계;
- [d5] 단계는 상기 정지조건을 만족하지 않는 경우 상기 정지조건을 만족할 때 까지 상기 [d2] 단계 내지 상기 [d4] 단계를 반복하는 단계;
- [d] 단계 이후 [a] 단계 내지 [c] 단계를 반복

알고리즘

[a] 신경망을 이용 로봇 궤적 계산

- [a1] 로봇의 현재와 이전 위치 차를 이용 로봇 자세 변화값, 특징점의 위치 현재와 이전 시점 **신경망 입력**
- [a2] 신경망에서 로봇의 자세 변화에 대한 오차로 로봇의 **자세 변화 값 수정**
- [a3] 로봇의 자세 변화 값 오차로 보정된 **현재 위치 계산**
- [a4] 오차가 보정된 **로봇의 궤적 계산**

[b] 계산된 궤적을 바탕으로 로봇이 관측한 특징점들의 지도를 작성

하나의 **특징점**을 여러 위치에서 중복되어 관측 어느 정도 흩어져서 **분포**
 센서 시스템 **잡음**에 의해 관측된 특징점의 그 위치가 어느 정도 흩어져서 측정된
 로봇 시스템 **잡음**이 더해져 주행하면 잡음 모델이 더해져서 결정

[c] 작성된 지도를 평가하여 지도 작성을 종료 할 것인지 판단

특징점 지도에서 특징점들을 **Clustering** 분산 총합 계산
 소정의 기준값과 비교하여 **지도작성 종료**할 것 인지 보다 **정확한 로봇의 궤적** 계산할 것인지를 **판단**

[d] 작성된 지도를 평가한 결과 소정의 기준을 만족하지 않는
 경우 진화연산 기법을 사용하여 신경망 내부의 노드들의 연결 상태를 조절

신경망을 진화 연산에 적합한 염색체의 형태로 변환시킴

신경망은 7 개의 입력 노드로 구성된 **입력층**
 연결 강도 값들로 이루어진 행렬
 은닉층에 대한 입력 벡터 u
 실행함수

다수의 은닉 노드로 구성된 **은닉층**
 은닉층 출력 벡터 h
 연결 강도 값들로 이루어진 행렬
 출력층 입력 벡터 o
 실행함수

3 개의 출력 노드로 구성된 **출력층**
 신경망 출력 벡터 e

로봇이 $x(t-1)$ 에서 $x(t)$ 로 이동할 때 발생하는 오차를 보정

- [d1] 염색체를 N개 생성 **초기화**
- [d2] 돌연변이 2 N개의 염색체를 **생성**
- [d3] 로봇의 궤적을 계산 특징점 클러스터들 분산 총합으로 **평가**
- [d4] 새로운 N개의 염색체 **선택**
- [d5] 소정의 정지조건 만족 여부 **판단**

최종적인 로봇의 지도로서 출력

[d : [d5] 정지조건 만족할 때 까지 [d2], [d4] 반복] 이후 [a], [c] 반복