

지능성 발현을 위한 진화적응 하드웨어

정덕진, 이종호, 이필규, 이주환, 조성배 / 인하대, 연세대

머리말

근대 산업혁명 이후 약 200 여년 동안, 과학기술 및 산업문명은 엄청난 속도로 발전을 거듭하여 왔다. 특히, 2차 세계대전을 겪으면서 현대 과학기술은 전기, 전자, 기계 기술을 중심으로 발전되었고, 20세기 최고의 기술인 디지털 컴퓨터의 도래로 정보통신기술이 모든 기술의 핵심이 되고 있으며, 미래 인간문명의 기반을 이루고 있다. 이러한 과학기술은 수 세기동안 결정론적 세계관을 바탕으로 발전되었으며, 그 패러다임은 스스로 주변 환경에 적응 및 진화하는 지능적 기능을 포함하지 못하고 있다.

그러나, 21세기의 첨단 과학시대의 기술들은 자율적으로 적응 및 진화하는 기능을 필수적으로 요구하게 될 것이다. 특히, 고도의 정보화 사회에서 필수라 할 수 있는 대규모 실시간 정보처리 (Large-scale real-time information processing)를 위해서는 지능적이고 효율적이며 적재 적소에서 다양한 요구에 응할 수 있는 지능형 컴퓨터 기술이 필요하게 된다. 이러한 컴퓨터 기술의 핵심 기술은 주변 환경을 실시간에 감지/인식 (Sensing/Perception)하고 감지된 상황에

따라 실시간 예측/판단/계획(Prediction /Decision/ Planning)하며 변화하는 환경에 적응하여 임무를 정확히 수행(Performance)하는 기술이다.

진화라고 하는 것은 기본적으로 유전적인 변화를 수반하는 자기 복제의 과정이다. 진화연산은 진화의 원리를 이용하여 어려운 문제들을 해결하는 새로운 컴퓨터 과학의 접근법으로서 누구도 미리 알고 있거나 지시하지 않은 프로그램이나 지능이 스스로 자생적으로 만들어지는 '창발성'을 갖는 것이 가장 큰 매력이다. 이를

이용하여 인공 생명을 만들어내거나 프로그램이 프로그램을 짤거나 다양한 데이터들로부터 유용한 숨은 규칙들을 찾아낸다거나 하는 작업 등을 하기도 한다. 진화에는 예정된 길이 있는 것이 아니고, 적응도에 따른 피선택/생존의 확률에 의하여 적자생존(survival of the fittest)의 법칙만이 유일한 가이드가 되며 생명의 본질인 번식/재생산(reproduction)의 궤도에서 돌연변이와 교배에 의한 변이를 겪으며 다양성을 자산으로 삼아 보다 나은 형태와 습성을 모색하게 된다

진화형 하드웨어

(EHW: Evolvable Hardware)

가) 디지털하드웨어의 진화

궁극적으로 하드웨어 진화는 변화하는 정보에 의존해 하드웨어의 구조를 자율적으로 바꾸는 것을 말한다. 기존의 하드웨어는 디자이너가 그 목적에 따라 정교하게 고안한 함수를 논리회로로 표현하고 여기에 타이밍과 기타 수반될 수 있는 복잡한 제약들을 고려하는 과정을 통해 디자인된다. 이렇게 제작된 하드웨어는 설계시 고려된 상황이 계속 유지된다면 예정된 우수한 성능을 발휘하겠지만, 만일 설계시 미처 예기치 못한 환경의 변화가 생겼을 경우 무기력해지거나 오동작을 일으킬 수 있다. 예를 들어 숙련되지 않은 디자이너가 설계한 회로의 입력단에 비정상적인 입력이 들어오면 그 하드웨어는 예상대로의 동작을 하지 못하게 된다. 행성 탐사 우주선이 예상치 못한 난기류나 우주선, 지형의 특이성으로 인해 제어체계가 오동작을 일으키는 예도



소특집 ①

있을 것이다. 요컨대 현재까지의 하드웨어는 설계단계부터 광범위한 사전지식과 상황예측이 필요하고, 설계시에 예상치 못하는 상황이 발생할 경우 구조가 고정된 하드웨어로서는 기능의 회복이 불가능할 수 있다. 이는 우주공간과 같이 하드웨어가 동작될 환경에 대한 사전 지식이 불충분하거나, 디자인 시 종종 무시되는 사소한 요인이 의외로 전체에 큰 영향을 끼칠 수 있는 복잡한 시스템의 경우, 특히 전문가의 감시와 대처가 불가능할 때 발생가능한 상황이다. 이러한 진화형 하드웨어는 1960년대부터 논의되어 왔지만, 진화형 하드웨어의 필수 조건인 결선이나 배선, 단위기능의 조합이 재구성 가능해야 한다는 요구조건을 잘 반영한 FPGA가 등장하고 유전자 알고리즘등이 구체화된 최근에 이르러서야 비로소 그 효용성을 인정받기 시작했다.

- 재구성이 가능한 소자 'FPGA'

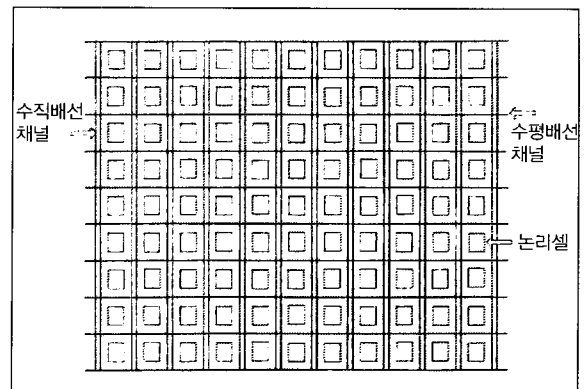
초기의 진화 하드웨어는 진화된 회로의 크기가 작다는 공통적인 문제를 가지고 있었다. 하드웨어진화는 AND 게이트나 OR 게이트와 같은 기초적인 게이트를 기본으로 하고 있었다. 이런 하위레벨 형태에서의 진화 상태를 게이트 레벨(gate-level)진화라고 불렀는데, 이 진화는 산업적인 응용에서 큰 효용이 없었다. 그러나 비약적인 발전을 할 수 있는 계기가 있었는데, 바로 'FPGA'의 등장이었다. 이로서 게이트 수준에서 벗어나 기능 레벨(function-level)의 새로운 형태의 하드웨어 진화를 연구할 수 있게 되고, 그 응용범위 또한 넓힐 수 있었다.

- 'FPGA (Field Programmable Logic Array)'란 무엇인가?

위에서 언급한 하드웨어를 진화를 구현하는데 있어서 하드웨어의 유연성, 즉 임의로 재구성이 가능한 것은 매우 중요하다. 이것을 적용하기 위해서는 하드웨어가 하나의 셀 단위로 이뤄져서 협조 행동이 이뤄지도록 유도해야한다. 현재 연구중인 대부분의 진화하드웨어는 FPGA를 기반으로 이루어지고 있다.

'FPGA'는 AND면과 OR면 등 2단 어레이 구조를 가지는 PLD(Programmable Logic Device)의 일종으로 PLD를 보다 확장시킨 형태이다. FPGA는 아래 그림에서 알 수 있는 것처럼 논리 셀들이 2차원적으로 배

열된 형태이다. 이 그림에서 조그만 사각형이 한 개의 논리 셀을 나타낸다. 그리고 이들 셀의 상하, 좌우로 배선 공간(즉, 배선 채널)이 놓여 있다. 이 배선 채널을 통해 해당 FPGA의 입출력 핀과 연결된다.



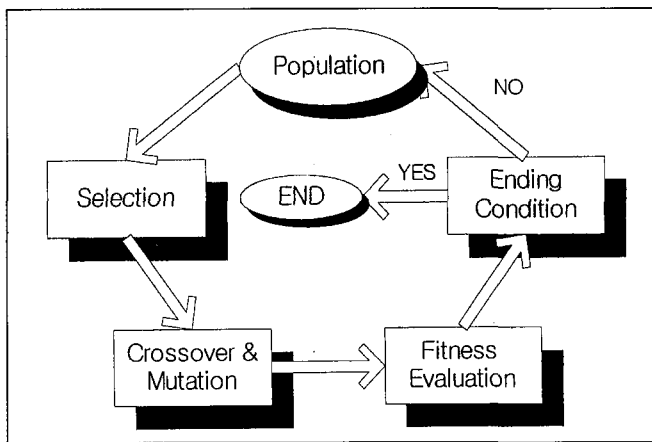
나) 유전자 알고리즘

유전자 알고리즘(Genetic Algorithm : GA)은 진화 하드웨어를 진화시키기 위해 가장 많이 사용되는 알고리즘이다. 이것은 유전학과 자연진화를 흉내낸 적응 탐색법으로써 1975년 Holland에[1]의해 개발된 것이다. 복잡한 최적화 문제를 해결하기 위해서 GA는 집단을 사용하고 여기에 모의진화를 일으켜 이를 점진적으로 개선해 나가게 된다. 집단은 다수의 염색체로 형성되고 염색체는 문제공간 상의 한점(잠정적인 해)을 대표하게 된다. 집단을 유지함으로써 여러방향으로 탐색을 추구하게 된다. 흔히 염색체들은 유전자의 역할을 반영하도록 비트 스트링 형태로 표현된다.

그림 2는 GA의 기본 구조를 보여주고 있다. 먼저 초기 집단이 형성되고 염색체들의 강점과 약점이 적합도 항목으로 평가된다. 이 때 각 염색체는 복호화되어 목적함수(이득함수)를 제공하고 이로부터 적합도가 계산된다. 재생산(reproduction)은 적합도값에 따라 더 적합한 개체들을 선택하여 다음 세대의 집단을 형성하고, 교배(crossover)는 개체들 사이의 정보교환이 가능하게 하며, 돌연변이(mutation)는 유전자를 임의로 변경하여 집단에 변화를 도입한다. 이와 같이 한 세대 동안 재생산, 교배, 돌연변이를 거치면서 새롭게 형성된 집단은

다시 평가되고 앞서 수행한 일련의 연산과정은 최적의 해가 발견될 때까지 반복된다.

이 방법은 일반적인 탐색 알고리즘에 비하여 전역 탐색능력이 상대적으로 우수하다고 알려져 있으며 진화형 하드웨어의 주된 도구로 사용되고있는 최적화 기법이다. 유전자 알고리즘의 구조는 대체로 다음의 순서로 기술될 수 있다.



다) 진화형 하드웨어(EHW)란 무엇인가?

진화 하드웨어(Evolvible Hardware: EHW)는 미지의 환경에 적응하여 스스로 구성을 변형시킬 수 있는 하드웨어이다. 하드웨어 시스템의 탄생(설계제작)에서부터 환경변화에 따른 적응이 회로자체에서 이루어지는 과정에 이르기까지 EHW는 회로의 기능과 구조를 사전계획하여 설계하는 일반적인 하드웨어 회로와 대비하여 본질적인 차이를 나타내게 된다. 위에 설명된 'FPGA'는 비트스트링을 가해 줌으로써 내부의 회로구조가 결정된다. 즉, 가해진 비트스트링이 달라지면 내부 구조 또한 달라지는 것이다. 이를 이용하여 작게는 게이트 레벨 진화에서 크게는 FPGA 구조결정 비트스트링을 유전자 알고리즘의 염색체로 이용한다. 이 염색체는 FPGA의 회로 구조를 결정하고 환경으로부터 평가를 받는다. 이 평가 결과를 토대로 선택과 유전자 연산(crossover, mutation)을 수행해 다음 세대의 개체군을 형성한다. 세대를 반복하면서 환경에 최적의 결과를 내는 하드웨어 구조가 생성된다. 이를 모듈 단위로 확

대하면 즉, 유전자 알고리즘으로 기능 모듈사이의 결선이나 배치 등을 결정함으로써 시스템을 구성한다면 복잡한 구조의 회로를 설계할 수 있다.

이와 같은 장점을 가진 진화 하드웨어를 사용하면 복잡한 전문가 지식과 중앙컨트롤을 통해서나 수행할 수 있었던 컨트롤 작업을 진화 하드웨어의 환경 적응 능력으로 대체할 수 있을 뿐만 아니라, 변화가 심한 복잡한 환경 속에서 최적의 상태를 스스로 유지하는 효과를 얻을 수 있다. 그러나, 진화 하드웨어는 아직 개발 초기 단계의 기술로서 그 실용의 한계가 많이 남아 있다.

라) 기술 동향

진화형 하드웨어에 대한 제1회 워크샵이 1995년 스위스에서 개최된 이래로 많은 연구자들의 성과가 발표되어왔다. 최초로 intrinsic EHW의 가능성을 보여준 영국의 A. Thompson과 Zebulum[2]은 1KHz와 10KHz입력을 구분하는 Tone Discriminator를 FPGA상에서 성공적으로 진화 시켰고, 장애물 회피능력을 가진 로봇 컨트롤러의 진화도 보였다. 일본의 Higuchi[3]가 이끄는 연구진은 EHW의 상업적인 응용에 있어서 선두를 달리고 있는데, 이들은 적응형 의수제어 컨트롤러, 영상압축, 신경망과 결합된 적응형 Equalizer등에서 좋은 연구결과들을 발표하고 있다. 스위스의 Sipper[4]는 FireFly라는 실험적인 intrinsic on-line EHW를 FPGA로 구현하였고, 현재 "embryonics"라 하여 스스로 성장하고 재생산하며 오류를 수정하는 생명체의 특성을 구현한 재구성 하드웨어를 개발중에 있다. Stoica[5]는 Gaussian 신호 발생기의 진화나 우주 공간에서의 적응형 센서에 대한 연구를 수행했고, J. Miller와 P. Thomson[6]은 EHW의 탐색능력을 활용하여 기존의 ALU를 능가하리라 여겨지는 최적의 디자인을 찾는 작업을 계속해왔고, 지금까지 특허로 등록된 많은 디자인을 재발견하는 성과를 얻었다. 현재도 이들은 특허등록되지 않은 디자인을 찾아내는 작업을 계속하고 있다. De Garis[7]는 최근까지 일본의 ATR에서 CAM-brain 프로젝트에 참가하여 3차원 Cellular Automata에 기반하는 대규모 진화신경망을 구성하여 인공두뇌를 구현하는 연구를 계속해왔다.

진화 하드웨어 개발

가) 간단한 논리회로의 진화

유전자 알고리즘을 이용하여 하드웨어가 특정목적을 수행할 수 있도록 그 결선과 로직의 구성을 변화시켜 진화하는 경우를 살펴보자. 이러한 진화에 있어 하드웨어의 구성에 관한 정보를 유전자 알고리즘의 염색체(chromosome)로 표현해야하는 작업이 선행된다. 하드웨어 진화에 많이 사용되고 있는 FPGA는 그 결선 및 로직의 구성을 bit-string으로 표현한다. 또한, 유전형질들이 원하는 방향으로 진화하기 위한 적합도 함수도 정의해야 한다. 초기에 임의로 생성된 염색체는 각각 독립된 개체로 존재하고 적합도 함수에 의해 우월형질로 평가된 것은 높은 선택될 확률을 갖고 교배에 의해 우월형질을 다음세대로 넘겨주게 된다. 아래 그림3은 Xilinx사의 XC6216 FPGA로 간단한 전 가산기를 진화를 통하여 구성한 예를 보인다. 본 실험에서는 어떤 외

부의 도움 없이 또한 바람직한 입출력쌍 이외의 다른 사전 지식이 없는 상태에서도 하드웨어가 진화하여 가산기가 구성되었다.

나) 진화하드웨어를 이용한 결함극복회로

또 다른 진화형 하드웨어의 응용사례로서 재구성 하드웨어에서의 결함극복회로를 구현하였다. 여러 가지 하드웨어의 결함극복대책에도 불구하고 집적회로의 용량이 커가면서 복잡도가 증가하여 하드웨어의 신뢰도가 중요한 이슈가 된다.

전통적인 결함극복회로는 구성요소의 완전한 대체품을 필요로 하므로 하드웨어의 낭비가 심했다. 이를 극복하기 위해 재구성하드웨어를 기반으로 한 다양한 시도가 있어왔으나, 발생한 결함에 대한 정확한 파악이 가능할 때만 결함극복이 가능하다는 공통된 제약을 안고 있다. 진화하드웨어의 자율적응특성을 적용하여 이를 해결할 수 있다는 가능성이 제기된 바 있으나, 게이트 레벨에서의 진화특성상 적응도를 정량적으로 평가하기가 어렵고 대규모회로의 재구성시 발생하는 전역탐색시간 및 성공률의 문제로 인해 구현이 어려웠다.

진화형 하드웨어를 두 개의 여분(dual redundancy)을 갖도록 구현한 경우를 생각해보자(그림4). 모듈1이 동작 중에 고장나면 모듈2가 이를 대체하고, 모듈2가 동작을 대행하는 동안 모듈1에선 재진화를 통해 결함을 극복한다. 이후, 모듈2가 고장나면 기존의 결함극복 시스템은 수명을 다하지만, 재진화를 이용한 시스템에

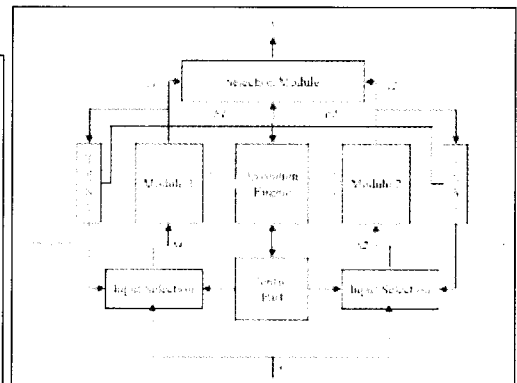
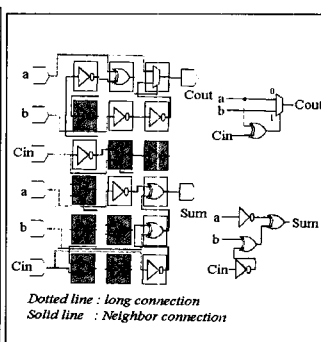
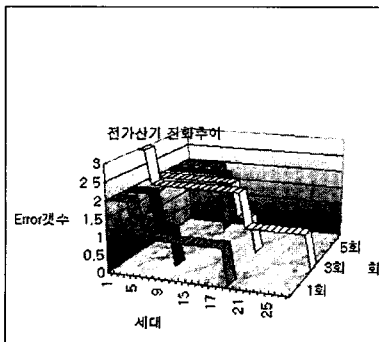
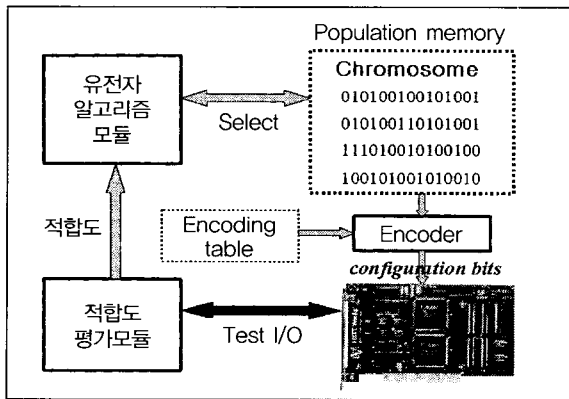


그림 3 Gate-level의 Intrinsic EHW 개념도 및 결과

그림 4 두개의 여분을 가지는 진화형 하드웨어의 개념도

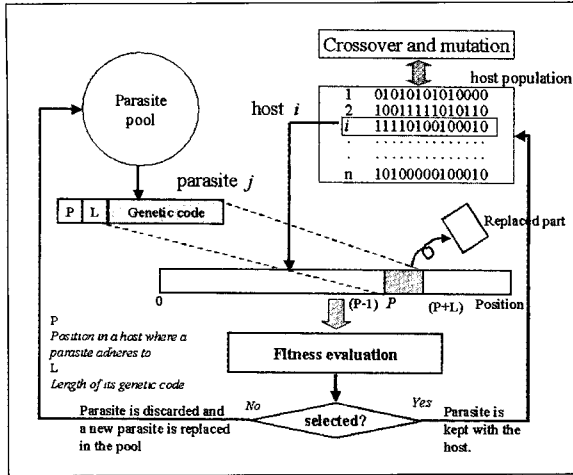


그림 5 CPGA의 개념

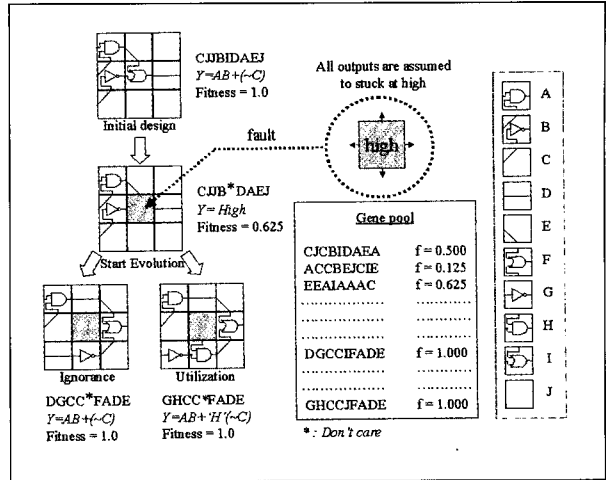


그림 6 EHW에 의한 결함극복 원리

선 수리된 모듈이 다시 기능을 수행함으로써 시스템의 동작을 유지한다.

만약 고장난 모듈을 수리하는데 걸리는 시간인 MTTR(Mean Time To Repair)이 대체동작하는 상대편 모듈이 고장나는데 걸리는 시간인 MTTF(Mean Time To Failure)보다 짧다면 이 시스템은 동일한 여분의 개수를 갖는 전통적인 결함 극복시스템에 비해 높은 활용도(availability) 시스템을 구축하게 한다. 이러한 재진화에 의한 결함 극복시스템을 위해 gate-level에서의 전역 해 탐색에 적합한 새로운 알고리즘(CPGA: Cooperative Parasite Genetic Algorithm)을 그림 5에 나타내었다.

CPGA(Cooperative Parasite GA)는 적응 도평가를 통해 생존한 Host에 기생한 parasite에게만 자기복제를 허용함으로써 parasite의 유전자풀이 Host의 적응도 향상에 기여하는 다양한 부분정보로 구성되도록 유도한다. 제안한 방법에선 parasite의 자기복제가 host의 유전자풀에서 발생하는 진화기작과 독립적으로 수행되어 Parasite의 유전자풀 자체는 자기복제된 유전형이나 완전히 새로운 유전형으로 채워지므로, 적당히 우수한 유전형이 유전자풀로 급속하게 파급되어 해의 다양성을 파괴하는 현상이 일어나지 않고, 국소해로 수렴해버린 host의 유전자풀에 parasite가 끊임없이 새로운 유전형을 제공하여 전역탐색을 가능케한다.

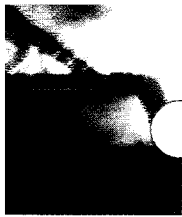
진화형 결함극복회로는 FPGA상에 구현된 전자회

로를 고유한 입출력을 갖는 작은 타일의 결합으로 간주하여 부분의 유지보수를 통해 전체의 동작을 보장하는 J. Lach의 제안을 발전시킨 것으로서, 각 타일을 EHW로 구현함으로써 결함이라는 환경변화에 대한 자율적응능력을 부여하였다. 기존의 결함극복회로에서 반드시 필요한 결함분석 및 대안구조의 사전설계는, 별도의 시스템 자원을 소모할 뿐만아니라 발생가능한 모든 결함을 사전에 예측하고 이에 대응하는 완전한 예비구조를 고안-저장하는 과정이 매우 소모적이고 항상 가능한 것도 아니다. EHW기반의 결함극복회로에선 해당 과정이 불필요하므로 하드웨어 자원을 절약하고 사전지식없이 결함에 자율 적응한다(그림).

SRAM기반의 FPGA상에서 실제로 구현된 EHW기반 결함극복회로는 4비트 소수 검출회로를 구현한 타일에 대해 적용한 결함극복실험에서, 영구적인 결함과 일시적인 결함, 단수 및 복수개의 결함발생시 결함의 특성이나 플랫폼인 FPGA의 구조에 대한 사전정보 없이 스스로 발생한 결함을 회피하거나 포용함으로써 기능을 회복하였다.

맺는 글

앞으로 진화 연산의 응용 가능성은 무궁 무진하다. 대표적인 예가 사람과 같은 기능을 가지는 인공 생명체를 만들어 내는 것이다. 실제로 공간에서 먹이를 적절



소특집 ①

히 분산시켜 놓고 이들을 효율적으로 찾는 인공 개미에 관한 실험에서 진화연산은 사람이 전혀 상상을 못했던 탁월한 재미를 만들어 내었다. 스케줄링 문제들도 진화 연산의 대표적 적용 대상이다. 여러 개의 도시를 가장 짧은 거리를 여행하면서 모두 방문하는 순회 세일즈맨 문제는 이러한 스케줄링 문제들의 원형으로 컴퓨터 과학에서 가장 어려운 문제 중의 하나로 꼽힌다. 유전 알고리즘으로 318 도시를 운행하는 문제까지는 최적의 해답을 찾아 놓고 있는 상태다.

기존의 하드웨어는 사람이 그 기능을 미리 정하고 설계를 마치면 일생 동안 정해진 작업을 반복한다. 이제 이 하드웨어에 진화의 기능을 넣는 작업들이 진행되고 있다. 컴퓨터 칩이 장착된 후에 계속 상황에 맞게 기능을 바꾸어 가며 성장하는 모습을 상상해 보라. 이 상상으로부터 이루어지던 일이 이미 초보적인 단계에서 실현되고 있다.

하드웨어라는 용어는 소프트웨어에 대한 상대적인 개념으로 컴퓨터 또는 전자회로의 개발과정에서 말 그대로 딱딱한 형태와 딱딱한 기능을 수행하는 부품을 의미하고자 사용되었다. 그러나 근래에 와서 이 둘의 대비개념은 상당부분 이완되어 *firmware*, *middleware*, *flexiware* 등의 중도적 신용어가 대두되었는가 하면 인공의 생물학적 기능소자를 의미하는 *wetware*까지 등장하면서 하드웨어의 명확한 정의를 흐리게 하고 있다. 또한 기능 면에서도 하드웨어 부품과 소프트웨어 *function*의 역할은 시스템 내에서 서로 대체가능하며 기능적 보완을 위한 *hardware-software codesign* 기술이 도입되면서 예전과 같은 기능분담의 개념은 점차 사라지고 있다. 이와 더불어 재구성형 회로(*reconfigurable circuit*)의 출현에 따라 하드웨어의 자동합성 또는 실리콘 컴파일레이션과 작동중 개선(*run-time reconfiguration*) 가능한 회로가 *FPGA*로 구현되고 있다. 이와 같이 하드웨어가 점차 구조상의 유연성을 획득하면서 전에 상상할 수 없었던 기능을 발휘하게 되는데 이러한 예가 진화형 하드웨어라 할 수 있다.

이 진화형 하드웨어는 실세계의 데이터와 효과적으로 교류하며 주변환경을 인식하고 적응학습하여 인간과 함께 지식과 정보를 축적함으로써 인간의 판단을 보조하고 활동을 보완하는 수단으로 쓰여지게 될 것이다. 또한, *bio*와 *digital*과 *mobile*로 대변되는 첨단 과학기

술 분야의 한 축으로서 수퍼지능기술은 미래사회의 지능정보화의 필수적인 요소로 자리잡게 될 것이다.

Referance

[1] J. H. Holland, *Adaptation in Natural and Artificial Systems*. University of Michigan Press, 1975.

[2] A. Thompson, P. Layzell, and R. S. Zebulum, *Exploration in design space : unconventional electronics design through artificial evolution*. Tr. IEEE on Evolution Computation, vol.3, no.3, pp.167-169, 1999.

[3] T. Higuchi, M. Murakawa, M. Iwata, I. Kajitani, W. Liu, and M. Salami, *Evolvable hardware at function level*, Pr. IEEE Int. InterMag' 97 Magnetics conference, 1997.

[4] Moshe Sipper, Maxime Goeke, Daniel Mange, Andre Stauffer, Eduardo Sanchez, Maroc Tomassini, *The Firefly machine: Online Evolvable*, Proceedings of the 1997 IEEE International InterMag '97 Magnetics Conference, , 1997.

[5] Adrian Stoica, Didier Keymeulen, Carlos-Salazar Lazaro, Wei-te Li, Ken Hayworth, and Raoul Tawel, *Toward On-board Synthesis and Adaptation of Electronic Functions: An Evolvable Hardware Approach*, Proceedings of the 1999 IEEE Aerospace Conference - Volume 2, pp. 351-357, 1999.

[6] J. F. Miller and P. Thompson, *Aspects of Digital Evolution: Geometry and Learning*, ICES98, Lecture notes in Computer Science, Springer-Verlag pp. 25-35, 1998.

[7] Hugo de Garis, Norberto Eiji Nawa, Michael Hough, Michael Korokin, *Evolving an Optimal De/Convolution Function for the Neural Net Modules of ATR's Artificial Brain Project*, Proceedings of the International Joint Conference on Neural Networks - Volume 1, pp.438-443, 1999.

[8] 산업자원부 주관 '수퍼지능칩 및 응용기술 개발' 중간보고서 (1, 2차년도)