(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2009-0012773
(43) 공개일자 2009년02월04일

(51) Int. Cl.[9]
G06F 9/50 (2006.01) *G06F 17/10* (2006.01)
(21) 출원번호 10-2007-0076903
(22) 출원일자 2007년07월31일
심사청구일자 2007년07월31일

(71) 출원인
고려대학교 산학협력단
서울 성북구 안암동5가1 고려대학교 내
(72) 발명자
이상근
서울 동대문구 제기2동 벽산아파트 105동 1001호
변은정
서울 성북구 안암동5가 고려대학교
(뒷면에 계속)
(74) 대리인
김인한, 김희곤

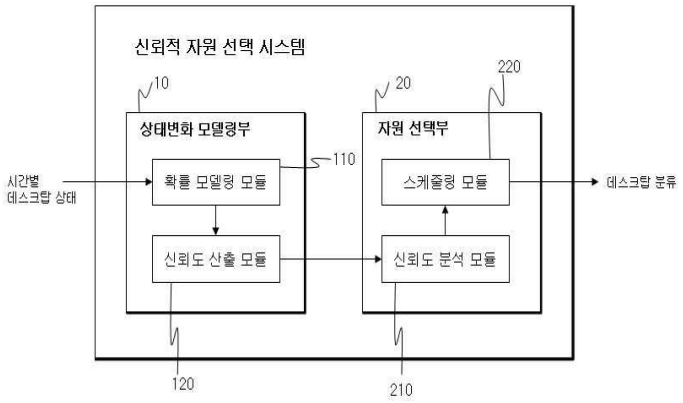
전체 청구항 수 : 총 12 항

(54) 데스크톱 상태의 은닉 마코브 모델 기반 신축 윈도우를이용한 신뢰적 자원 선택 시스템 및 그 방법

(57) 요 약

본 발명은 데스크톱 상태의 은닉 마코브 모델 기반 신축 윈도우를 이용한 신뢰적 자원 선택 시스템 및 그 방법에 관한 것으로, 더욱 상세하게는 데스크톱 자원의 상태 변화를 은닉 마코브 모델을 이용하여 확률 모델로 모델링하고, 상기 모델링을 통해 시스템은 성격 및 패턴을 유추하고, 자원의 기여에 대한 질적 평가를 적용하여 각 시간대에 적합한 자원을 선출함은 물론, 작업의 양에 부합하는 자원을 선택하는 신뢰적 자원 선택 시스템 및 그 방법을 제공하는 데 있다. 특히, 본 발명은 계산 집약적 응용에서 각각 다른 계산 능력을 요구하는 응용의 필요에 적합한 자원을 선택하는 것을 특징으로 한다.

대 표 도 - 도1



(72) 발명자

최성진

서울 성북구 안암동5가 고려대학교

김홍수

서울 성북구 안암동5가 고려대학교

특허청구의 범위

청구항 1

데스크톱의 상태 변화를 은닉 마코브 모델을 이용하여 확률 모델로 모델링을 수행하여 신뢰도를 계산하는 상태 변화 모델링부와;

상기 상태변화 모델링부로부터 모델링된 확률 모델 상에 신축 윈도우(EW)를 이용하여 동적인 데스크톱의 상태 변화를 예측하여 안정적이고 신뢰적인 자원을 선택하는 자원 선택부를 포함하는 것을 특징으로 하는 데스크톱 상태의 은닉 마코브 모델 기반 신축 윈도우를 이용한 신뢰적 자원 선택 시스템.

청구항 2

제 1항에 있어서,

상기 상태변화 모델링부는

일정한 시간 단위로 체크하여 데스크탑의 상태를 유휴상태(I), 사용상태(U), 정지상태(S)로 구분하고, 시간 단위별 상태 변화를 은닉 마코브 모델을 이용하여 확률로 계산하여 모델링을 수행하는 확률 모델링 모듈과;

비터비 알고리즘을 적용하여 각 시간 단위(time unit)에서 나타날 데스크톱 상태의 상태 신뢰도를 계산하는 신뢰도 산출 모듈을 포함하는 것을 특징으로 하는 데스크톱 상태의 은닉 마코브 모델 기반 신축 윈도우를 이용한 신뢰적 자원 선택 시스템.

청구항 3

제 2항에 있어서,

상기 확률 모델링 모듈은

하기의 수식과 같은 은닉 마코브 모델을 이용하여 모델링을 수행하는 것을 특징으로 하는 데스크톱 상태의 은닉 마코브 모델 기반 신축 윈도우를 이용한 신뢰적 자원 선택 시스템.

$$Model = \{N, M, A, B, \pi\}$$

$$Q = \{ q_1, q_2, \cdots, q_N \}$$

$$O = \{ o_1, o_2, \cdots, o_M \}$$

$$a_{ij} = P(q_{t+1} = j \mid q_t = i)$$

$$b_j(k) = P(O_t = k \mid q_t = j), \quad i = k = M$$

$$\pi = P[q_1 = i], \quad 1 = i = M$$

여기서, N, M은 스칼라 값으로 N=48, M=3, A, B, π는 확률분포 값이다.

청구항 4

제 2항에 있어서,

상기 신뢰도 산출 모듈은

은닉 마코브 모델의 확률 모델 상에서 하기의 수식을 이용하여 신뢰도를 산출하는 것을 특징으로 하는 데스크톱 상태의 은닉 마코브 모델 기반 신축 윈도우를 이용한 신뢰적 자원 선택 시스템.

$$SC = \sum_{i=1}^{N} \sum_{j=1}^{N} \left(\alpha_t(i) \bullet a_{i,j} \bullet b_j(O_{t+1}) \bullet \beta_{t+1}(j) \right)$$

여기서, α와 a의 곱의 합은 전방향 상태 신뢰도(FSC)이며, β와 b의 곱의 합은 후방향 상태 신뢰도(BSC)이다.

청구항 5

제 1항에 있어서,

상기 자원 선택부는

상기 확률 모델 상에 신축 윈도우(EW)를 이용하여 윈도우를 확장 또는 축소를 통해 윈도우 크기를 결정하고, 오른쪽으로 이동하면서 상태 신뢰도(SC_{Ii})를 계산하여 작업에 필요한 상태 신뢰도(SC_{need})를 만족하는지 여부를 검사하는 신뢰도 분석모듈과;

만족하는 경우 측정된 상태 신뢰도(SC_{min}, SC_{avg}, SC_{max})를 바탕으로 유연하고 맞춤화된 자원 선택 방식을 제공하는 스케줄러 모듈을 포함하는 것을 특징으로 하는 데스크톱 상태의 은닉 마코브 모델 기반 신축 윈도우를 이용한 신뢰적 자원 선택 시스템.

청구항 6

제 5항에 있어서,

상기 신뢰도 분석모듈은

검사 시작점은 주어진 시간대를 기준으로 최좌측에 위치시키고, 이를 기점으로 좌우측에 시간제한을 위한 경계점을 각각 체크하고, 검사시작점과 시간제한점들 간에 윈도우를 위치시키고 윈도우 내의 상태 신뢰도(SC_{Ii})를 계산하는 것을 특징으로 하는 데스크톱 상태의 은닉 마코브 모델 기반 신축 윈도우를 이용한 신뢰적 자원 선택 시스템.

청구항 7

제 5항에 있어서,

상기 신뢰도 분석모듈은

데스크톱의 상태 신뢰도 값이 증가된 경우 또는 작업이나 사용자의 요구에 의해 제공 시간양이 증가한 경우에는 윈도우 크기를 확장하고,

데스크톱의 상태 신뢰도 값이 감소한 경우 또는 작업이나 사용자의 요구에 의해 제공 시간양이 감소한 경우에는 윈도우 크기를 축소하는 것을 특징으로 하는 데스크톱 상태의 은닉 마코브 모델 기반 신축 윈도우를 이용한 신뢰적 자원 선택 시스템.

청구항 8

제 5항에 있어서,

상기 스케줄러 모듈은

상기 신뢰도 분석모듈의 상태 신뢰도 분석결과 각 데스크톱의 상태 신뢰도(SC_{Ii})가 최소 상태 신뢰도(SC_{min})보다 크거나 같은 경우 우등 데스크톱으로 분류하고,

각 데스크톱의 상태 신뢰도(SC_{Ii})가 최소 상태 신뢰도(SC_{min})보다 작거나 같고, 평균 상태 신뢰도(SC_{avg}) 보다 크거나 같은 경우 온건 데스크톱으로 분류하고,

각 데스크톱의 상태 신뢰도(SC_{Ii})가 평균 상태 신뢰도(SC_{avg})보다 작거나 같고, 최대 상태 신뢰도(SC_{max}) 보다 크거나 같은 경우 열등 데스크톱으로 분류하고,

각 데스크톱의 상태 신뢰도(SC_{Ii})가 최대 상태 신뢰도도(SC_{max}) 보다 작거나 같은 경우 부적당 데스크톱으로 분류하여 각 분류된 종류의 데스크톱에 각각 다른 종류의 작업을 할당하여 신뢰적인 연산 수행이 가능한 것을 특징으로 하는 데스크톱 상태의 은닉 마코브 모델 기반 신축 윈도우를 이용한 신뢰적 자원 선택 시스템.

청구항 9

(a) 데스크톱의 상태 변화를 은닉 마코브 모델을 이용하여 확률 모델로 모델링을 수행하여 신뢰도를 계산하는 단계와;

(b) 상기 모델링된 확률 모델 상에 신축 윈도우(EW)를 이용하여 동적인 데스크톱의 상태 변화를 예측하여 안정적이고 신뢰적인 자원을 선택하는 단계를 포함하는 것을 특징으로 하는 데스크톱 상태의 은닉 마코브 모델 기반

신축 윈도우를 이용한 신뢰적 자원 선택 방법.

청구항 10

제 9항에 있어서,

상기 (a) 단계는

일정한 시간 단위로 체크하여 데스크탑의 상태를 유휴상태(I), 사용상태(U), 정지상태(S)로 구분하고, 시간 단위별 상태 변화를 은닉 마코브 모델을 이용하여 확률로 계산하여 모델링을 수행하는 단계와;

상기 확률 모델 상에 비터비 알고리즘을 적용하여 각 시간 유닛에서 나타날 데스크톱 상태의 상태 신뢰도(SC)를 계산하는 단계를 포함하는 것을 특징으로 하는 데스크톱 상태의 은닉 마코브 모델 기반 신축 윈도우를 이용한 신뢰적 자원 선택 방법.

청구항 11

제 9항에 있어서,

상기 (b) 단계는

(b1)상기 확률 모델 상에 신축 윈도우(EW)를 이용하여 윈도우를 확장 또는 축소를 통해 윈도우 크기를 결정하고, 오른쪽으로 이동하면서 상태 신뢰도(SC_{Ii})를 계산하여 작업에 필요한 상태 신뢰도(SC_{need})를 만족하는지 여부를 검사하는 단계와;

(b2)만족하는 경우 측정된 상태 신뢰도(SC_{min}, SC_{avg}, SC_{max})를 바탕으로 유연하고 맞춤화된 자원 선택 방식을 제공하는 단계를 포함하는 것을 특징으로 하는 데스크톱 상태의 은닉 마코브 모델 기반 신축 윈도우를 이용한 신뢰적 자원 선택 방법.

청구항 12

제 11항에 있어서,

상기 (b2) 단계는

상기 신뢰도 분석모듈의 상태 신뢰도 분석결과 각 데스크톱의 상태 신뢰도(SC_{Ii})가 최소 상태 신뢰도(SC_{min})보다 크거나 같은 경우 우등 데스크톱으로 분류하고,

각 데스크톱의 상태 신뢰도(SC_{Ii})가 최소 상태 신뢰도(SC_{min})보다 작거나 같고, 평균 상태 신뢰도(SC_{avg}) 보다 크거나 같은 경우 온건 데스크톱으로 분류하고,

각 데스크톱의 상태 신뢰도(SC_{Ii})가 평균 상태 신뢰도(SC_{avg})보다 작거나 같고, 최대 상태 신뢰도(SC_{max}) 보다 크거나 같은 경우 열등 데스크톱으로 분류하고,

각 데스크톱의 상태 신뢰도(SC_{Ii})가 최대 상태 신뢰도도(SC_{max}) 보다 작거나 같은 경우 부적당 데스크톱으로 분류하여 각 분류된 종류의 데스크톱에 각각 다른 종류의 작업을 할당하는 것을 특징으로 하는 데스크톱 상태의 은닉 마코브 모델 기반 신축 윈도우를 이용한 신뢰적 자원 선택 방법.

명 세 서

발명의 상세한 설명

기 술 분 야

<1> 본 발명은 데스크톱 상태의 은닉 마코브 모델 기반 신축 윈도우를 이용한 신뢰적 자원 선택 시스템 및 그 방법에 관한 것으로, 더욱 상세하게는 데스크톱 자원의 상태 변화를 은닉 마코브 모델을 이용하여 확률 모델로 모델링하고, 상기 모델링을 통해 시스템은 성격 및 패턴을 유추하고, 자원의 기여에 대한 질적 평가를 적용하여 각 시간대에 적합한 자원을 선출함은 물론, 작업의 양에 부합하는 자원을 선택하는 신뢰적 자원 선택 시스템 및 그 방법을 제공하는 데 있다. 특히, 본 발명은 계산 집약적 응용에서 각각 다른 계산 능력을 요구하는 응용

의 필요에 적합한 자원을 선택하는 것을 특징으로 한다.

<2> 본 발명은 신축 윈도우를 이용하여 신축 연산과 확장 연산의 동작 방식 및 자원 선정 방식 등을 제공한다. 본 발명은 특정 자원이 주어진 조건과 작업의 성격에 부합하는 지를 판단하는 중요한 척도를 제공하고, 유연하고 맞춤화된 자원 선택 방식을 제공한다. 결과적으로 본 발명은 더욱 공여도가 크고 안정적인 자원을 선택할 수 있고, 시간적 작업 종료 제약이 있는 응용 등에 적용될 수 있다.

배 경 기 술

<3> 그리드 컴퓨팅 시스템은 대규모의 응용을 다수의 분산 자원을 이용하여 구현한 플랫폼이다. 데스크톱 그리드 컴퓨팅 시스템은 보다 유연한 패러다임으로써, 안정적인 고사양의 슈퍼컴퓨터나 고가의 장비를 사용하는 기존의 그리드와 달리 열등하고 안정성이 부족한 데스크톱 자원들을 이용하여 고성능, 고처리력을 제공하고 있다. 데스크톱 그리드는 LAN이나 인터넷 등의 다양한 네트워크 환경을 기반으로 연결된 다양한 다수의 이질적인 데스크톱들로 구성된다. 게다가, 이러한 혁신적인 패러다임은 저비용으로 대규모 계산 능력을 형성하는 발명의 주도적인 역할을 해왔으며, 주로 기상 관측, 우주 연구, 분자 구조 연구, 의학 관련 연구 등의 많은 계산 능력을 요구하는 대용량의 응용을 수행하고 있다.

<4> 그러나, 전통적인 그리드 환경에서 성능은 주로 응용 간의 자원 공유로 인해 불규칙해 지는데 반해, 데스크톱 그리드에서는 데스크톱 자원의 사용자활동이나, 소유자의 성향에 의해 시스템이 동적이고, 불안정해진다. 심지어 몇몇 극단적 참여 속성 유형의 사용자들은 자원 제공 및 기여에 대해 비 예측적으로 행동하여 시스템의 신뢰성을 저하시킨다.

<5> 이처럼 거대한 컴퓨팅 풀을 구성하기 위해 인터넷 기반의 네트워크 구조는 다수의 데스크톱 자원을 취합하는데 도움을 주지만, 한편으로는 데스크톱의 휘발적이고 동적 속성은 시스템의 안정성과 신뢰성의 위협하는 불안한 요소로 자리 잡고 있다. 특히, 휘발적인 자원을 배제하지 않는 데스크톱 그리드 시스템의 경우, 불안정한 자원 기여 및 비 예측적 연산 수행으로 인해 성능 저하와 연산 수행 시간이 지연되는 문제가 있었다.

<6> 따라서, 예측을 지원하는 스케줄링 기법은 시스템의 성능과 자원 유용성을 향상하는 한편, 적절한 스케줄링 정책을 결정할 수 있도록 도움을 준다. 다수의 경험 기반의 연구들은 과거의 선험적 정보를 기반으로 시스템의 성능을 예측해왔다. 그러나 기존 연구들이 접근하였던 예측의 방법은 대부분 통계적 방식들을 이용하였다. 기존 연구들은 자원의 동적 상태 변화를 분석하고 불확실성과 변화에 대한 문제를 해결하고자 노력해 왔으나, 그 방법의 한계로 인해 전반적 경향이나 전체적인 성격, 요소들 간의 상호연관성 등을 파악하는데 그치는 것을 알 수 있다. 보다 정확한 예측과 강력한 의견 결정을 위해 시스템은 반드시 데스크톱 자원의 동적 속성, 휘발성을 고려해야 하며 적합한 분석적 접근에 적응해야만 하는 기술적 과제를 안고 있다.

발명의 내용

해결 하고자하는 과제

<7> 본 발명은 상기와 같은 문제점을 해결하기 위해 안출된 것으로써, 본 발명의 목적은 데스크톱 그리드가 기존의 그리드와 자원의 성격, 즉 자원의 안정성에 있어 그 차이점이 있다는 점에 착안하여, 스케줄링 시에 이러한 자원의 휘발성 및 자율적 자원 기여 패턴을 고려하고, 특히 자원의 휘발성에 기인해 발생하는 연산 완료 지연 및 시스템 안정성 저하를 해결하기 위해 안정적 자원을 선출하여 작업을 할당할 수 있는 신뢰적 자원 선택 시스템 및 그 방법을 제공하는 데 있다.

과제 해결수단

<8> 본 발명은 데스크톱의 휘발성(volatility)로 인한 연산수행의 안정성 저하와 시스템 성능 저하 방지를 위해 데스크톱의 상태의 변화를 예측하기 위해 시간의 흐름에 따라 변동되는 데스크톱의 상태를 보다 정교하게 예측하기 위해 데스크톱의 상태 변화를 은닉 마코브 모델을 이용하여 확률 모델링하고, 확률 모델 상에 비터비 알고리즘을 적용하여 각 시간 유닛에서 나타날 데스크톱 상태의 최확치를 제공하고, 확률 모델 상에 신축 윈도우(EW)를 적용하여 사용자의 요구나 시스템의 상태 변화에 따라 유연하게 상태 검사 시점을 조절하고, 구해진 최확치에 대해 가중치 등을 부과하여 신뢰도 값을 더욱 정교하게 만드는 것을 특징으로 한다.

<9> 본 발명에 따른 데스크톱 상태의 은닉 마코브 모델 기반 신축 윈도우를 이용한 신뢰적 자원 선택 시스템은 데스크톱의 상태 변화를 은닉 마코브 모델을 이용하여 확률 모델로 모델링을 수행하여 신뢰도를 계산하는 상태변

화 모델링부와 상기 상태변화 모델링부로부터 모델링된 확률 모델 상에 신축 윈도우(EW)를 이용하여 동적인 데스크톱의 상태 변화를 예측하여 안정적이고 신뢰적인 자원을 선택하는 자원 선택부를 포함하는 것을 특징으로 한다.

<10> 여기서, 상기 상태변화 모델링부는 일정한 시간 단위로 체크하여 데스크탑의 상태를 유휴상태(I), 사용상태(U), 정지상태(S)로 구분하고, 시간 단위별 상태 변화를 은닉 마코브 모델을 이용하여 확률로 계산하여 모델링을 수행하는 확률 모델링 모듈과 비터비 알고리즘을 적용하여 각 시간 단위(Time unit)에서 나타날 데스크톱 상태의 상태 신뢰도를 계산하는 신뢰도 산출 모듈을 포함하는 것을 특징으로 한다.

<11> 그리고, 상기 확률 모델링 모듈은 하기의 수식과 같은 은닉 마코브 모델을 이용하여 모델링을 수행하는 것을 특징으로 한다.

$$Model = \{N, M, A, B, \pi\}$$

$$Q = \{ q_1, q_2, \cdots, q_N \}$$

$$O = \{ o_1, o_2, \cdots, o_M \}$$

$$a_{ij} = P(q_{t+1} = j \mid q_t = i)$$

$$b_j(k) = P(O_t = k \mid q_t = j), \quad i = k = M$$

$$\pi = P[q_1 = i], \quad 1 = i = M$$

<12>

<13> 여기서, N, M은 스칼라 값으로 N=48, M=3이며 A, B, π는 확률분포 값이다.

<14> 또한, 상기 신뢰도 산출 모듈은 은닉 마코브 모델의 확률 모델 상에서 하기의 수식을 이용하여 신뢰도를 산출하는 것을 특징으로 한다.

$$SC = \sum_{i=1}^{N} \sum_{j=1}^{N} \left(\alpha_t(i) \bullet a_{i,j} \bullet b_j(O_{t+1}) \bullet \beta_{t+1}(j) \right)$$

<15>

<16> 여기서, α와 a의 곱의 합은 전방향 상태 신뢰도(FSC)이며, β와 b의 곱의 합은 후방향 상태 신뢰도(BSC)이다.

<17> 그리고 상기 자원 선택부는 상기 확률 모델 상에 신축 윈도우(EW)를 이용하여 윈도우를 확장 또는 축소를 통해 윈도우 크기를 결정하고, 오른쪽으로 이동하면서 상태 신뢰도(SC_{Ii})를 계산하여 작업에 필요한 상태 신뢰도(SC_{need})를 만족하는지 여부를 검사하는 신뢰도 분석모듈과 만족하는 경우 측정된 상태 신뢰도(SC_{min}, SC_{avg}, SC_{max})를 바탕으로 유연하고 맞춤화된 자원 선택 방식을 제공하는 스케줄러 모듈을 포함하는 것을 특징으로 한다.

<18> 또한, 상기 신뢰도 분석모듈은 검사 시작점은 주어진 시간대를 기준으로 최좌측에 위치시키고, 이를 기점으로 좌우측에 시간제한을 위한 경계점을 각각 체크하고, 검사시작점과 시간제한점들 간에 윈도우를 위치시키고 윈도우 내의 상태 신뢰도(SC_{Ii})를 계산하는 것을 특징으로 한다.

<19> 그리고, 상기 신뢰도 분석모듈은 데스크톱의 상태 신뢰도 값이 증가된 경우 또는 작업이나 사용자의 요구에 의해 제공 시간 양이 증가한 경우에는 윈도우 크기를 확장하고, 데스크톱의 상태 신뢰도 값이 감소한 경우 또는 작업이나 사용자의 요구에 의해 제공 시간 양이 감소한 경우에는 윈도우 크기를 축소하는 것을 특징으로 한다.

<20> 또한, 상기 스케줄러 모듈은 상기 신뢰도 분석모듈의 상태 신뢰도 분석결과 각 데스크톱의 상태 신뢰도(SC_{Ii})가 최소 상태 신뢰도(SC_{min})보다 크거나 같은 경우 우등 데스크톱으로 분류하고, 각 데스크톱의 상태 신뢰도(SC_{Ii})가 최소 상태 신뢰도(SC_{min})보다 작거나 같고, 평균 상태 신뢰도(SCavg)보다 크거나 같은 경우 온건 데스크톱으로 분류하고, 각 데스크톱의 상태 신뢰도(SC_{Ii})가 평균 상태 신뢰도(SC_{avg})보다 작거나 같고, 최대 상태 신뢰도(SC_{max}) 보다 크거나 같은 경우 열등 데스크톱으로 분류하고, 각 데스크톱의 상태 신뢰도(SC_{Ii})가 최대 상태 신뢰도도(SC_{max}) 보다 작거나 같은 경우 부적당 데스크톱으로 분류하여 각 분류된 종류의 데스크톱에 각각 다른 종류의 작업을 할당하여 신뢰적인 연산 수행이 가능한 것을 특징으로 한다.

<21> 한편, 본 발명에 따른 데스크톱 상태의 은닉 마코브 모델 기반 신축 윈도우를 이용한 신뢰적 자원 선택 방법은 (a) 데스크톱의 상태 변화를 은닉 마코브 모델을 이용하여 확률 모델로 모델링을 수행하여 신뢰도를 계산하는 단계와 (b) 상기 모델링된 확률 모델 상에 신축 윈도우(EW)를 이용하여 동적인 데스크톱의 상태 변화를 예측하

여 안정적이고 신뢰적인 자원을 선택하는 단계를 포함하는 것을 특징으로 한다.

<22> 여기서, 상기 (a) 단계는 일정한 시간 단위로 체크하여 데스크탑의 상태를 유휴상태(I), 사용상태(U), 정지상태(S)로 구분하고, 시간 단위별 상태 변화를 은닉 마코브 모델을 이용하여 확률로 계산하여 모델링을 수행하는 단계와 상기 확률 모델 상에 비터비 알고리즘을 적용하여 각 시간 유닛에서 나타날 데스크톱 상태의 상태 신뢰도(SC)를 계산하는 단계를 포함하는 것을 특징으로 한다.

<23> 그리고, 상기 (b) 단계는 (b1)상기 확률 모델 상에 신축 윈도우(EW)를 이용하여 윈도우를 확장 또는 축소를 통해 윈도우 크기를 결정하고, 오른쪽으로 이동하면서 상태 신뢰도(SC_{Ii})를 계산하여 작업에 필요한 상태 신뢰도(SC_{need})를 만족하는지 여부를 검사하는 단계와 (b2)만족하는 경우 측정된 상태 신뢰도(SC_{min}, SC_{avg}, SC_{max})를 바탕으로 유연하고 맞춤화된 자원 선택 방식을 제공하는 단계를 포함하는 것을 특징으로 한다.

<24> 여기서, 상기 (b2) 단계는 상기 신뢰도 분석모듈의 상태 신뢰도 분석결과 각 데스크톱의 상태 신뢰도(SC_{Ii})가 최소 상태 신뢰도(SC_{min})보다 크거나 같은 경우 우등 데스크톱으로 분류하고, 각 데스크톱의 상태 신뢰도(SC_{Ii})가 최소 상태 신뢰도(SC_{min})보다 작거나 같고, 평균 상태 신뢰도(SC_{avg}) 보다 크거나 같은 경우 온건 데스크톱으로 분류하고, 각 데스크톱의 상태 신뢰도(SC_{Ii})가 평균 상태 신뢰도(SC_{avg})보다 작거나 같고, 최대 상태 신뢰도(SC_{max}) 보다 크거나 같은 경우 열등 데스크톱으로 분류하고, 각 데스크톱의 상태 신뢰도(SC_{Ii})가 최대 상태 신뢰도도(SC_{max}) 보다 작거나 같은 경우 부적당 데스크톱으로 분류하여 각 분류된 종류의 데스크톱에 각각 다른 종류의 작업을 할당하는 것을 특징으로 한다.

효 과

<25> 상기에서 살펴본 바와 같이, 본 발명에 따른 데스크톱 상태의 은닉 마코브 모델 기반 신축 윈도우를 이용한 신뢰적 자원 선택 시스템 및 그 방법은 자원의 기여를 예측하기 힘든 역동적 데스크톱 그리드 환경에서 보다 안정적이고 검증된 자원을 선정하는 진보된 자원 선택 기법을 제안함으로써, 연산 수행의 안정성을 제공하고, 연산의 완료성을 향상시킬 수 있는 탁월한 효과가 발생한다.

<26> 또한, 본 발명은 작업 완료를 위해 필요한 시간 요구사항을 충족하는 적절한 데스크톱을 선정하고, 주어진 작업이나 사용자의 요구 사항에 부합하여 시스템이 유연하게 적응할 수 있는 탁월한 효과가 발생한다.

<27> 그리고, 본 발명은 데스크톱 그리드 컴퓨팅 환경에서 스케줄링 시에 데스크톱의 동적, 자율적 연산 수행 특성을 고려하기 위해 데스크톱의 가용성, 즉 연산 수행 패턴에 대한 마코브 체인을 이용한 확률 모델링을 제안하여 예측의 신뢰성을 향상시키고, 작업의 크기나 종류에 따라 적합한 데스크톱을 선정하여 데스크톱의 연산 유형을 보다 정확하게 예측할 수 있는 탁월한 효과가 발생한다.

<28> 이상에서 설명한 본 발명의 상세한 설명에서는 본 발명의 바람직한 실시예를 참조하여 설명하였지만, 본 발명의 보호범위는 상기 실시예에 한정되는 것이 아니며, 해당 기술분야의 통상의 지식을 갖는 자라면 본 발명의 사상 및 기술영역으로부터 벗어나지 않는 범위 내에서 본 발명을 다양하게 수정 및 변경시킬 수 있음을 이해할 수 있을 것이다.

발명의 실시를 위한 구체적인 내용

<29> 이하, 본 발명의 구체적인 구성 및 작용에 대하여 도면 및 실시예를 참조하여 상세하게 설명하기로 한다.

<30> 도 1은 본 발명의 바람직한 실시예에 따른 신뢰적 자원 선택 시스템의 블럭도이다. 도 1을 참조하면, 본 발명에 따른 신뢰적 자원 선택 시스템은 상태변화 모델링부(10)와 자원 선택부(20)를 포함하여 구성될 수 있다.

<31> 상기 상태변화 모델링부(10)는 확률 모델링 모듈(110)과 신뢰도 산출 모듈(120)을 포함하여 구성될 수 있으며, 상기 자원 선택부(20)는 신뢰도 분석 모듈(210)과 스케줄링 모듈(220)을 포함하여 구성될 수 있다.

<32> 보다 구체적으로, 본 발명에 따른 신뢰적 자원 선택 시스템의 상태변화 모델링부(10)의 확률 모델링 모듈(110)은 데스크톱의 시간 변화에 따른 상태 변화를 표현하기 위해 각 데스크톱의 상태를 30분마다 검사하여 기록하며, 하루를 주기로 데스크톱의 상태 변화를 모델링한다. 따라서 하루는 총 48개의 시간 단위로 모델링되고, 각 시간 단위의 상태와 시간 단위 간의 상태 전이 확률을 각각 계산한다.

<33> 데스크톱의 상태는 유휴 상태 I (idle), 사용 상태 U (use), 정지 상태 S (stop)로 구분된다. I 상태는 20분 이

상 연산 수행 도중에 사용자의 간섭 없이 연산을 수행할 수 있는 상태이다. U 상태는 사용자의 일시적 간섭이나 자원 점유로 인해 연산을 수행할 수 없는 상태이다. S는 자원의 물리적 결함(crash)이나 시스템 종료, 네트워크 단절로 연산을 수행할 수 없는 상태이다.

<34> 데스크톱의 가용 상태는 매 30분마다 검사하여 자원의 시간대별 상태를 마코브 체인에 갱신하고 확률을 재계산한다.

<35> 도 1은 본 발명의 바람직한 실시예에 따른 데스크탑 자원의 상태를 유휴상태(I), 사용상태(U), 정지상태(S)로 분류하여 시간의 흐름에 따른 이들 간의 전의를 은닉 마코브 모델을 이용하여 상태 전이 예측을 모델링한 예시도이다.

<36> 도 1을 참조하면, 기본적인 마코브 체인은 도 1과 같이 모델링된다. 도 1에 나타난 바와 같이 I_i는 i번째 시간 단위의 I상태를 의미한다. 전이 확률 $P_{I_iI_j}$ 는 i번째 시간 단위의 I상태에서 j번째 시간 단위의 I상태로 전이하는 확률을 의미한다.

<37> 하기 <수학식1>의 전이 확률 행렬 P는 j시간 단위에서 k시간 단위로의 상태 전이 확률들을 표현한다.

수학식 1

<38>
$$P_{C_jC_k} = \begin{pmatrix} P_{I_jI_k} & P_{I_jU_k} & P_{I_jS_k} \\ P_{U_jI_k} & P_{U_jU_k} & P_{U_jS_k} \\ P_{S_jI_k} & P_{S_jU_k} & P_{S_jS_k} \end{pmatrix}$$

<39> 상태 전이 확률 행렬에서 각 행의 합은 1이 된다.

<40> 이와 같이 데스크톱의 각 시간대에 나타날 수 있는 상태들의 확률로 각 데스크톱의 각 시간대의 가용 상태를 예측할 수 있다.

<41> 상태변화 모델링부의 신뢰도 산출 모듈(120)은 예측의 정확도를 향상시키기 위해 마코브 모델 내의 각 상태의 신뢰도 값을 계산한다. 마코브 체인 모델에서 데스크톱의 가용성은 실제 연산을 수행할 수 있는 I 상태를 중심으로 I를 지나가는 모든 경로들의 확률 값을 통해 표현된다. I 상태의 신뢰도는 가용 상태를 표현한다. 또한 비터비(viterbi) 알고리즘을 적용하여 최확치 (most probable value) 및 최적 상태 열을 추출하여 시간 변화에 따른 상태 변화의 예측을 지원한다.

<42> I 상태를 지나가는 모든 경로의 확률은 전방향 상태 신뢰도 FSC (Forward State Credit)와 후방향 상태 신뢰도 BSC (Backward State Credit)의 곱으로 구할 수 있다.

<43> 하기의 <수학식 2>와 <수학식 3>은 각각 전방향 상태 신뢰도와 후방향 상태 신뢰도를 계산하는 방식을 나타내고 있다.

<44> 전방향 상태 신뢰도는 이전 시간대에서 현재 I 상태로 전이하는 모든 상태의 전이 확률과 이전 상태의 신뢰도 값의 곱들의 합으로써 <수학식2>와 같이 계산된다. 후방향 상태 신뢰도는 현재 I 상태에서 다음 시간대의 상태로 전이하는 확률과 다음 상태의 신뢰도의 곱들의 합으로써 <수학식3>과 같이 계산된다.

수학식 2

<45>
$$FSC_I_i = SC_I_{i-1} \cdot P_{Ii-1Ii} + SC_U_{i-1} \cdot P_{Ui-1Ii} + SC_S_{i-1} \cdot P_{Si-1Ii}$$

수학식 3

<46>
$$BSC_I_i = SC_I_{i+1} \cdot P_{IiIi+1} + SC_U_{i+1} \cdot P_{IiUi+1} + SC_S_{i+1} \cdot P_{IiSi+1}$$

<47> 그리고, 하기의 <수학식 4>에 기술된 바와 같이 현재 I 상태를 중심으로 구해진 전방향 상태 신뢰도와 후방향 상태 신뢰도의 곱이 I의 상태 신뢰도 SC (State Credit)가 된다.

수학식 4

$$SC_I_i = FSC_I_i \cdot BSC_I_i$$

<48>

<49> 결론적으로 SC_I_i 값은 I_i를 대표하는 I_i의 신뢰도 값이 된다.

<50> 마코브 모델 내에서 처음 상태의 전방향 상태 신뢰도와 마지막 상태의 후방향 상태 신뢰도는 하기의 <수학식 5>와 같이 각각 일어나는 빈도의 비율로서 구한다. 예를 들어, 도 1에서 I_1, U_1, S_1 의 빈도를 각각 f_I_1, f_U_1, f_S_1 이라고 하면, I_1 의 전방향 신뢰도는 f_I_1, f_U_1, f_S_1 의 총합에 대한 f_I_1의 비율 값이다.

수학식 5

$$FSC_I_1 = \frac{f_I_1}{f_I_1 + f_U_1 + f_S_1}, \quad BSC_I_{48} = \frac{f_I_{48}}{f_I_{48} + f_U_{48} + f_S_{48}}$$

<51>

<52> 이하, 본 발명에 따른 신뢰적 자원 선택 시스템(SCREW, Selection ofCredible Resource with Elastic Window)이 어떻게 동작하는지 상세히 기술한다.

<53> SCREW는 확률적 상태 변화 모델에 기반하고 있으며 통계적 접근 방식이 아닌 확률적 접근방식을 이용함으로써, 시간의 변화에 따른 상태 변화를 보다 정확하게 이용할 수 있으며, 동적인 특성을 잘 반영할 수 있다.

<54> 비록 그동안 통계적 모델링 방식이 폭넓게 이용되어 왔고 저비용으로 패턴을 분류할 수 있지만, 시간에 따라 복잡하고 동적으로 변동되는 데스크톱 그리드 환경을 모델링하고 예측하기에는 불충분하다.

<55> 데스크톱 상태의 변화는 인자를 이용한 임의 과정(parametric random process)으로 모델링할 수 있으며, 확률적 변수의 인자들은 보다 높은 정확도로 측정될 수 있다. 제안된 방식은 상태를 인지하는 신뢰적 방식을 지원하고 인터넷 상의 동적인 데스크톱의 상태 변화를 예측한다.

<56> SCREW의 기반 모델은 확률적 접근 방식을 이용한 은닉 마코브 모델은 관측 인자로부터 은닉 인자를 결정하고 추출된 모델 인자는 응용을 분석하는데 이용된다. 은닉 마코브 모델에서는 각 상태가 가능한 출력들에 대한 확률 행렬을 가지고, 은닉 마코브 모델이 데스크톱 상태열에 대한 정보를 생성하고 있어, 상태는 직접적으로 가시적이지 않다.

<57> 은닉 마코브 모델은 각 상태를 고려하고, 상태 간의 전이에 다한 확률을 계산하고, 일별 갱신을 지원한다. 각 데스크톱의 상태는 매 반시간 간격 마다 점검되고 갱신된다. 데스크톱의 상태는 유휴 (I), 사용 (U), 정지 (S)의 세 가지 상태로 나뉜다. I 상태는 데스크톱 사용자의 개인적 활동이나 사용으로 인한 작업 수행 방해가 없어 데스크톱 그리드 응용을 중지 없이 수행할 수 있는 상태를 나타내며, U상태는 사용자의 사용이나 간섭, 일시적 수행 정지로 인해 응용을 수행하기 어려운 상태이다. S상태는 소프트웨어 결함이나 하드웨어, 기계 결함이나 네트워크 단절 등으로 인해 단위시간 동안 작업을 수행할 수 없는 상태를 의미한다.

<58> 하기의 <수학식 6>은 은닉 마코브 모델이 5개의 구성 성분(N, M,A,B, π)으로 모델링한 것을 나타낸다.

수학식 6

$$Model = \{N, M, A, B, \pi\}$$
$$Q = \{ q_1, q_2, \cdots, q_N \}$$
$$O = \{ o_1, o_2, \cdots, o_M \}$$
$$a_{ij} = P(q_{t+1} = j \mid q_t = i)$$
$$b_j(k) = P(O_t = k \mid q_t = j), \quad i = k = M$$
$$\pi = P[q_1 = i], \quad 1 = i = M$$

<59>

<60> 여기서, N, M은 스칼라 값이며, A, B, π는 확률분포 값이다.

<61> 보다 구체적으로, N은 모델 내에서의 상태의 수를 의미하며, 상기 <수학식 6>의 2열은 각각의 상태를 의미한다.

즉 q_1, q_2, …, q_N 들은 각각 다른 상태들을 나타낸다. 모델링된 마코브 모델에서는 각 시간 유닛이 은닉되어있는 상태들이며 총 개수 N는 48개 (24시간 * 2(반시간마다 모델링한다고 가정))가 된다.

<62> M은 이산적인 관측 심볼의 개수로서, 상기 <수학식 6>의 3열은 o_1, o_2, …, o_M 이 실제 관찰되는 상태들의 집합을 의미한다. 본 발명의 모델링된 은닉 마코브 모델에서는 관측되는 상태인 I(유휴), S(정지), U(사용)이 관측 상태들로서 M의 개수는 3이 된다.

<63> A는 A={a_{ij}}로 상태 천이 확률 행렬을 나타낸다. 상기 <수학식 6> 3열의 a_{ij}는 t시간의 상태 i가 t+1 시간의 j로 전이할 확률 $P(q_{t+1} = j \mid q_t = i)$ 을 나타낸다.

<64> B는 B={$B_j(k)$}로 관측 심볼 확률 분포이다. 상기 <수학식 6> 4열에서와 같이 시간 t에서 j상태에 있으면서 관측 심볼 K가 나타날 확률을 나타낸다. 즉, 특정 시간 유닛에서 특정 관측 심볼이 나타날 확률을 말한다. (예를 들어, 36시간 유닛에서 U 상태일 확률)

<65> π는 π_j로 초기 상태 분포를 나타낸다. 상기 <수학식 6> 5열에 나타난 바와 같이 초기시간에 I상태에 있을 확률 분포를 나타낸다. 이 분포에 따라 초기 상태를 선택한다.

<66> 또한, 독립적 관측열은 하기의 <수학식 7>과 같이 표현된다.

수학식 7

$$P(O \mid Q, \lambda) = \prod_{t=1}^{T} P(o_t \mid q_t, \lambda) = b_{q_1}(o_1) b_{q_2}(o_2) \ldots b_{q_T}(o_T), \quad \text{where } T=48$$

<67>

<68> 그리고, 최적의 상태열을 찾기 위해 전방향 상태 알고리즘, 후방향 상태 알고리즘을 이용하여 최측열을 제공하는 비터비 알고리즘(Viterbi algorithm)을 이용한다.

<69> 전방향 상태 알고리즘은 하기의 <수학식 7a>에 나타난 바와 같이 임의의 은닉 상태로 전이할 수 있는 이전의 상태와 각 이전 상태의 전이 확률 곱들의 합으로 구해질 수 있으며, 후방향 상태 알고리즘은 하기의 <수학식 7b>에 나타난 바와 같이 임의의 은닉 상태로 전이되는 다음 상태와 각 다음 상태의 전이 확률 곱들의 합으로 구해질 수 있다.

수학식 7a

$$\alpha_{t+1}(j) = \sum_{i=1}^{N} \left(\alpha_t(i) \bullet A_{i,j} \right) \bullet B_j(O_{t+1}), \quad \text{where } P(O \mid \lambda) = \sum_{r=1}^{k} a_T(n)$$

<70>

수학식 7b

$$\beta_t(i) = \sum_{j=1}^{N} \left(a_{i,j} \bullet b_j(O_{t+1}) \bullet \beta_{t+1}(j) \right)$$

<71>

<72> 또한, 상태열을 최대화시키는 $P(O,Q/\lambda)$ 를 비터비 알고리즘으로 디코드하기 위해 하기의 <수학식7c>을 사용한다. 하기의 <수학식 7-3>은 임의 상태에서 특정 관측열이 나타나도록 최대화시키는 값을 도출하는 식이다.

수학식 7c

$$\delta_{t+1}(i) = [\max_i (\delta_t(i) a_{ij})] b_j(o_{t+1})$$

<73>

<74> 그리고, 하기의 <수학식7d>는 은닉 마코브 모델 λ에서 특정 관측 상태가 발생할 확률을 나타내고, 하기의 <수학식7e>는 전방향 알고리즘과 후방향 알고리즘을 고려하여 임의의 상태에서 특정 관측 상태가 발생할 확률값인 신뢰도(SC)를 나타내며, 상기 <수학식4>의 정형적인 표현이다.

수학식 7d

$$P(O \mid \lambda) = \sum_{r=1}^{k} a_r(j)\beta_r(j) \quad \text{for any } j,\ 1 \le j \le n$$

<75>

수학식 7e

$$SC = \sum_{i=1}^{N} \sum_{j=1}^{N} \left(\alpha_t(i) \bullet a_{i,j} \bullet b_j(O_{t+1}) \bullet \beta_{t+1}(j)\right)$$

<76>

<77> 결국, 시간 단위에서 최측 상태 관측은 상태 신뢰도(SC)로 표현되고, 데스크톱의 가장 적합 상태를 나타내고 각 상태의 확률을 구한다.

<78> 데스크톱 기반의 플랫폼의 가장 큰 특징은 자원의 자율성, 휘발성 등이다. 안정적이고 신뢰적인 시스템 환경 제공을 위해서는 시스템은 반드시 이러한 불안정한 특성들을 고려해야 한다.

<79> 본 발명에 따른 신뢰도 자원 선택 시스템의 자원 선택부(20)는 휘발 자원에 대해 안정적 자원 선출을 위해 확률적 상태 전이 모델에 기반한 신축 윈도우 EW (elastic window)를 이용한다. 게다가, 신축 윈도우는 자가 조절 윈도로 윈도우를 축소, 확장하면서 연속 상태에 대한 연속 상태 신뢰도를 계산하여 각 상태가 출현할 확률을 나타낸다.

<80> 신축 윈도우 EW가 시간 기반 구조 및 전이 관계의 확률 관측 예측 모델을 기반으로 하고 있어 우수한 데스크톱 자원을 보다 신뢰적으로 선출할 수 있고, 작업의 특성이나 사용자의 요구에 적응하는 자원을 선택할 수 있다.

<81> 윈도우의 크기는 시간제한과 연관되어 있으며, 작업에 필요한 시간에 관련된 변수는 응용과 사용자 의해 제공되어야 한다.

<82> 상기 자원 선택부의 신뢰도 분석 모듈(210)은 데스크톱이 얼마나 오랫동안 양질의 연산을 수행할 수 있는지를 계산하기 위해 데스크톱의 사용가능한 윈도우를 계산한다. 이 신축 윈도우는 왼쪽에서 오른쪽으로 이동하면서 SC_{Ii} 를 계산하고 SC_{need} 와 비교하여 작업에 필요한 상태 신뢰도를 만족하는지 검사한다.

<83> 본 발명에 따른 신축 윈도우(EW)는 윈도우를 확장하거나 축소시키는 연산을 지원하는 유연한 구조를 가지고 있는 시간의 흐름에 적합한 구조이다. 윈도우 크기를 조절하기 위해, 상기 신뢰도 분석 모듈은 전방 이동, 후방 이동을 제공하며, 이들을 이용하여 검사 시간 및 기간을 정한다.

<84> 또한 주어진 작업에 필요한 시간양이나, 사용자가 요구하는 시간양을 기준으로 윈도우 크기를 지정하고, 윈도우 내의 상태 신뢰도를 검사하여, 주어진 시간 단위 동안 데스크톱이 얼마나 잘 작업을 수행할 수 있는지에 대한 정도를 파악하게 된다.

<85> 이를 위해, 신뢰도 분석 모듈(210)은 검사시작점은 주어진 시간대를 기준으로 최좌측에 위치시키고, 이를 기점으로 최우측에 시간제한을 위한 경계점을 각각 체크하고, 검사시작점과 시간제한점들 간에 윈도우를 위치시키고, 윈도우 내의 상태신뢰도를 계산한다. 윈도우를 확장시키는 경우는 두 가지가 있다.

<86> 첫째, 데스크톱의 안정적 연산 수행을 통한 상태신뢰도 값이 증가한 경우와 둘째, 작업이나 사용자의 요구에 의해 제공 시간양이 증가한 경우이다. 확장 연산에 의해 확대된 윈도우는 보다 넓은 시간 범위를 제공하게 된다.

<87> 한편, 윈도우 크기를 축소시키는 2가지 경우는 데스크톱의 상태 신뢰도가 감소되거나 요구되는 시간양이 감소할 경우이다. 축소 연산은 빈번히 휘발적 활동을 하는 데스크톱에 벌점을 부과하여 윈도우 크기를 줄인다. 윈도우의 최좌측 검사기준점은 검사시점에 고정되고, 시간기간을 정하는 지점은 우측으로 이동하며 시간 범위를 증가시키거나, 좌측으로 이동하여 시간 범위를 축소하게 된다.

<88> 즉, 왼쪽 기준점은 닫혀있고, 검사기준점은 오른쪽으로 열려있어 조건에 따라 윈도우의 크기를 조절한다. 윈도우가 오른쪽으로만 열린 것은 검사하려고 하는 부분이 임의의 시점부터 그 시점을 기준으로 미래의 시간에서 데스크톱의 상태를 점검하기 위한 것이다.

<89> 윈도우 기간의 마지막 지점은 기본 윈도우 기간의 마지막 지점으로 작업이 반드시 종료되어야 하는 시점을 나타

낸다.

<90> 신뢰도 분석 모듈(210)은 오른쪽으로부터 왼쪽으로 이동하면서 윈도우 크기를 축소시키고, 왼쪽에서 오른쪽으로 이동하면서 윈도우 크기를 증가시킨다. 만일 오른쪽 지점과 왼쪽 지점이 동일한 지점을 지정할 경우는 단순히 그 시간 단위의 상태 신뢰도만을 측정한다. 또한 이 경우는 현재 데스크톱 자원이 켜져 있는지 데스크톱 그리드 응용을 수행하고 있는지 점검할 수 있다.

<91> 도 3 ~ 도 5는 본 발명의 바람직한 실시예에 따른 확률 예측 모델을 기반으로 신축 윈도우가 어떻게 윈도우를 수축, 확장 시키는지의 과정과 상태 신뢰도를 구하는 과정을 나타낸다.

<92> 본 발명에 따른 자원 선택부의 신뢰도 분석 모듈(210)은 반드시 시작점부터 데드라인까지 데스크톱의 상태를 점검하면서 각 상태에 대한 상태신뢰도, 특히 데스크톱 응용을 수행할 수 있는 상태 신뢰도를 계산한다.

<93> 도 2에 나타난 바와 같이 확률 모델에서는 윈도우의 위치 변화는 고정된 좌측 시작점으로부터 우측 기준점들로 확장, 축소 연산을 이용하여 수행된다.

<94> 상기 신뢰도 분석 모듈(210)은 작업 수행에 필요한 주어진 상황에 맞는 상태신뢰도를 가질 때까지 이 과정을 반복하며 윈도우를 확장한다. 이 과정은 작업의 특성 및 요구 사항과 연관되어 윈도우의 확장 및 축소를 결정하고, 상태 신뢰도를 측정한다.

<95> 극단적인 경우에, 데스크톱은 빈번한 연산 중지와 자원 기여 중단, 물리적 능력 부족으로 인해 할당된 작업을 수행할 수 없게 된다. 따라서 측정된 상태신뢰도는 데스크톱가 얼마나 많은 양의 시간을 기여할 수 있는지, 주어진 시간 내에 작업을 수행할 수 있는지를 판단하는 기준이 되고, 자원 선택부의 스케줄러 모듈(220)은 신축 윈도우 EW 에 의해 측정된 상태 신뢰도 값을 바탕으로 작업을 할당하게 된다.

<96> 종래의 자원 선택 방법은 전형적으로 작업을 데스크톱이 작업을 요청한 순서대로 할당한다. 그러나, 본 발명에 따른 스케줄러 모듈(220)은 작업 할당 전에 데스크톱 자원이 주어진 작업을 수행하는데 적합한지를 검사한 후 작업을 할당하게 함으로써 보다 안정적인 연산환경을 제공하고 있다.

<97> 자원 선택은 각 응용의 각기 다른 요구에 따라 작업 할당 이전에 수행되어야 한다. 이러한 요구사항을 만족시키기 위해 본 발명은 데스크톱의 변화에 적응하여 보다 유연한 모델을 제공한다.

<98> 관측된 유한한 수의 데스크톱 상태 전이의 모델의 요소들을 가늠하기 위해 패턴과 경향들은 확률 모델을 기반으로 불규칙한 상태 변동으로부터 추출된다.

<99> 자원 선택부의 신뢰도 분석 모듈(210)은 신축 윈도우 EW를 이용하여 각 데스크톱들의 연산 수행 능력을 측정한다. 또한, 안정적 연산 수행 환경을 제공하고, 신뢰적인 자원 선택을 위해 윈도우의 크기를 조절한다. 윈도우의 크기를 결정하는 것은 작업 수행에 필요한 시간양이나 응용의 요구사항이다. 다음 시간대의 윈도우 크기는 이전 윈도우 크기에 부가적인 크기를 가감함으로써 결정된다.

<100> 윈도우의 크기 조절은 두 가지 분류된다. 첫째, 요구량 변화에 따른 조절이 있다. 둘째, 데스크톱의 자원제공에 대한 질을 고려하여 윈도우 크기를 조절할 수 있다.

<101> 하기의 <수학식 8>은 직전의 윈도우 크기에 작업에 필요한 실행 시간양의 차이를 고려하여 현재의 윈도우 크기를 고려하는 것을 나타낸다.

수학식 8

<102>
$$SW_{i+1} = SW_i \pm SW_{n_{TU}}$$

<103> 하기의 <수학식 9>는 데스크톱의 자원기여의 질을 고려하여 윈도우 크기를 조절하는 것을 나타낸다. 여기서 c는 상수 값으로 시스템의 상태에 따라 조절될 수 있는 값이며, ε는 데스크톱의 자원제공 패턴에 대한 불확실성과 예측의 정도를 나타내는 엔트로피 값으로 데스크톱이 얼마나 규칙적으로 자원을 기여하고 예측이 가능한지를 판단하도록 한다.

수학식 9

$$SW_{i+1} = SW_i - \left(\frac{SW_i + c\varepsilon}{SW_i}\right)$$

<104>

<105> 그러나 실제 검사 윈도우 크기가 데드라인보다 큰 경우는 적절한 데스크톱을 검색하기 난이하다. 따라서 검사 후에 유용한 데스크톱을 검색하지 못하는 결과를 초래하게 된다. 이로 인해 자원 검색으로 인한 시간낭비를 줄이기 위해, 신뢰도 분석 모듈(210)은 신축 윈도우 EW를 이용한 데스크톱 상태 검사 및 상태 신뢰도 검사 시에 각 데스크톱의 상태 신뢰도를 가지고 데스크톱 자원을 분류하게 된다.

<106> 상태 신뢰도 값은 응용의 특성에 따라 최소 상태 신뢰도(SC_{min}), 평균 상태 신뢰도(SC_{avg}), 최대 상태 신뢰도(SC_{max})로 각각 분류된다.

<107> 여기서, 최소 상태신뢰도(SC_{min})는 도 2에 나타난 바와 같이 첫 번째 종료 시점으로 작업이 요구하는 최소 완료 시간을 나타내며, 실제로 이 시간 기간은 작업의 최상의 종료 시간이다.

<108> 평균 상태 신뢰도(SC_{avg})는 도 3에 나타난 바와 같이 두 번째 종료시점이며 유효하게 인정되는 평균적인 실행 시간을 의미한다.

<109> 최대 상태 신뢰도(SC_{max})는 도 4에서 나타난 바와 같이 최종 데드라인, 즉 작업이 완료되어야 하는 최종 경계 시점이 된다. 만일 최종 상태 신뢰도를 넘어서면 더 이상 적합성 검사를 수행하지 않는다.

<110> 스케줄러 모듈(220)은 데스크톱의 상태 신뢰도 값과 이들을 각각 비교하여 데스크톱을 네 가지 종류로 분류한다. 이들은 각각 우등 데스크톱, 온건 데스크톱, 열등 데스크톱, 부적당 데스크톱이다. 이에 대한 자세한 설명은 하기의 도 5의 알고리즘 설명시 자세하게 기술하기로 한다.

<111> 분류된 종류의 데스크톱에는 각각 다른 정책을 적용하여 각각 다른 종류의 작업을 할당하여 신뢰적인 연산 수행을 수행하도록 한다.

<112> 우등 데스크톱의 경우 작업을 할당하여 작업을 수행하게 하고, 온건 데스크톱이나 열등 데스크톱에는 복제된 여분의 작업을 할당하여 작업의 완료성을 향상하고, 완료시간을 단축하도록 한다.

<113> 도 5는 본 발명의 바람직한 실시예에 따른 신축 윈도우 EW를 조절하여 자원을 선정하는 알고리즘을 도시한 것이다.

<114> 도 5를 참조하여 자원 선정 알고리즘에 대해 살펴보면 하기와 같다.

<115> ***Float SC_{Ii}, SC_{need}, SC_{min}, SC_{avg}, SC_{max}***

<116> //부동 소수점 형식으로 SC_{Ii}, SC_{need}, SC_{min}, SC_{avg}, SC_{max} 변수를 선언한다. SC_{Ii}는 은닉 마코브 모델의 i 은닉 상태에서 I 관측을 나타낼 상태 신뢰도 값이다. SC_{need}는 응용이 요구되는 상태신뢰도 값이다. SC_{min}, SC_{avg}, SC_{max}는 각각 최소, 평균, 최대 상태 신뢰도 값들로 구성되어 있고, 응용이나 시스템, 사용자로부터 입력되어 세팅된다.

<117> ***String DType***

<118> //문자열 Dtype 변수를 선언한다.

<119> ***Set WinStrPnt***

<120> //윈도우의 시작점을 설정한다. 즉 상태신뢰도를 검사할 시점을 지정한다.

<121> ***Calculate SC_{Ii}, SC_{need}***

<122> //SC_{Ii}, SC_{need}을 각각 계산한다. 즉 전방향과 후방향을 모두 이용하여 계산한다.

<123> ***Set EndPnt***

<124> //윈도우의 종료점을 설정한다. 즉 상태신뢰도 검사를 마칠 시점을 지정한다.

<125> SC_{need} = *Get* (SC_{min}, SC_{avg}, SC_{max})

<126> //SC_{min}, SC_{avg}, SC_{max}를 각각 입력 받는다.

<127> *Compare* (SC_{Ii}, SC_{need})

<128> //SC_{Ii}와 SC_{need}를 비교한다.

<129> *If* (SC_{Ii} >= SC_{min})
Mark DType=SuperiorDT

<130> //SC_{Ii} >= SC_{min} 이면 데스크톱의 타입은 우등 데스크톱으로 마크한다.

<131> *Elseif* (SC_{min} >= SC_{Ii} >= SC_{avg})
Mark DType=ModerateDT

<132> //SC_{min} >= SC_{Ii} >= SC_{avg} 이면 데스크톱의 타입은 온건 데스크톱으로 마크한다.

<133> *Elseif* (SC_{avg} >= SC_{Ii} >= SC_{max})
Mark DType=InferiorDT

<134> //SC_{avg} >= SC_{Ii} >= SC_{max}이면 데스크톱의 타입은 열등 데스크톱으로 마크한다.

<135> *Else* (SC_{Ii} <= SC_{max})
Mark DType=IncongruenDT

<136> //SC_{Ii} <= SC_{max} 이면 데스크톱의 타입은 부적당 데스크톱으로 마크한다.

<137> *Sort DType* (SC_{Ii});

<138> //데스크톱의 타입과 SC_{Ii}에 따라 데스크톱 리스트를 내림차순으로 정렬한다.

<139> *While* (WL_i){

<140> //할당할 작업 WL_i이 있을 동안 다음의 내용을 수행한다.

<141> *AllocateWL* (WL_i, *SuperiorDT*);

<142> //작업을 우등 데스크톱에 할당한다.

<143> *If* (*SuperiorDT* <= *RequiredDT*)

<144> //우등 데스크톱의 수가 작업 수행을 위해 요구되는 데스크톱의 수보다 작을 경우에는

<145> *AllocateWL* (WL_i, *ModerateDT*);

<146> //작업을 온건 데스크톱에 할당한다.

<147> *AllocateWL* (*Duplicate*(WL_i) , *InferiorDT*);

<148> //복제된 작업을 열등 데스크톱에 할당하여 작업의 복제본을 생성한다.

<149> WL_i++

<150> //다음 작업을 확인하다. 다음 작업이 있을 경우 while문을 재차 수행한다.

도면의 간단한 설명

<151> 도 1은 본 발명의 바람직한 실시예에 따른 신뢰적 자원 선택 시스템의 블럭도이다.

<152> 도 2는 본 발명의 바람직한 실시예에 따른 자원의 상태를 유휴 상태 I, 사용 상태 U , 정지 상태 S의 세가지 상태로 분류하여 시간의 흐름에 다른 이들의 간의 전이를 은닉 마코브 모델을 이용하여 상태 전이 예측을 모델링한 것이다.

<153> 도 3은 본 발명의 바람직한 실시예에 따른 신축 윈도우의 수축 연산과 확장 연산을 통한 상태 신뢰도의 계산 방식을 도시한 것이다.

<154> 도 4는 본 발명의 바람직한 실시예에 따른 신축 윈도우가 사용자가 지정한 임의의 중간 데드라인까지 확장하며 상태 신뢰도를 계산하는 방식을 도시한 것이다.

<155> 도 5는 본 발명의 바람직한 실시예에 따른 신축 윈도우가 마지막 데드라인까지 확장하며 상태 신뢰도를 계산하는 방식을 도시한 것이다.

<156> 도 6은 본 발명의 바람직한 실시예에 따른 신축 윈도우를 이용한 자원 분류와 자원 선택 알고리즘을 도시한 것이다.

<157> *도면의 주요부분에 대한 부호의 설명*

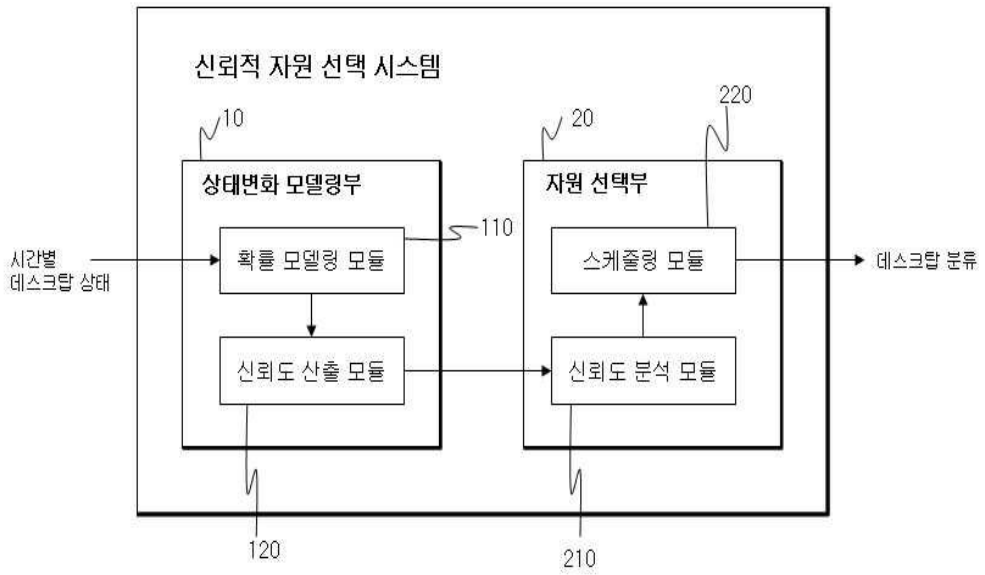
<158> 10 : 상태변화 모델링부 110 : 확률 모델링 모듈

<159> 120 : 신뢰도 산출 모듈 20 : 자원 선택부

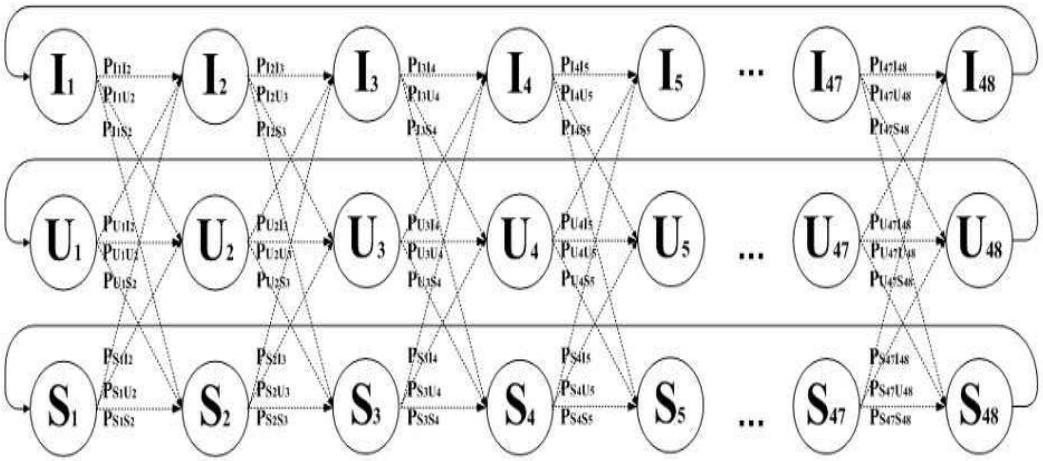
<160> 210 : 신뢰도 분석 모듈 220 : 스케줄러 모듈

도면

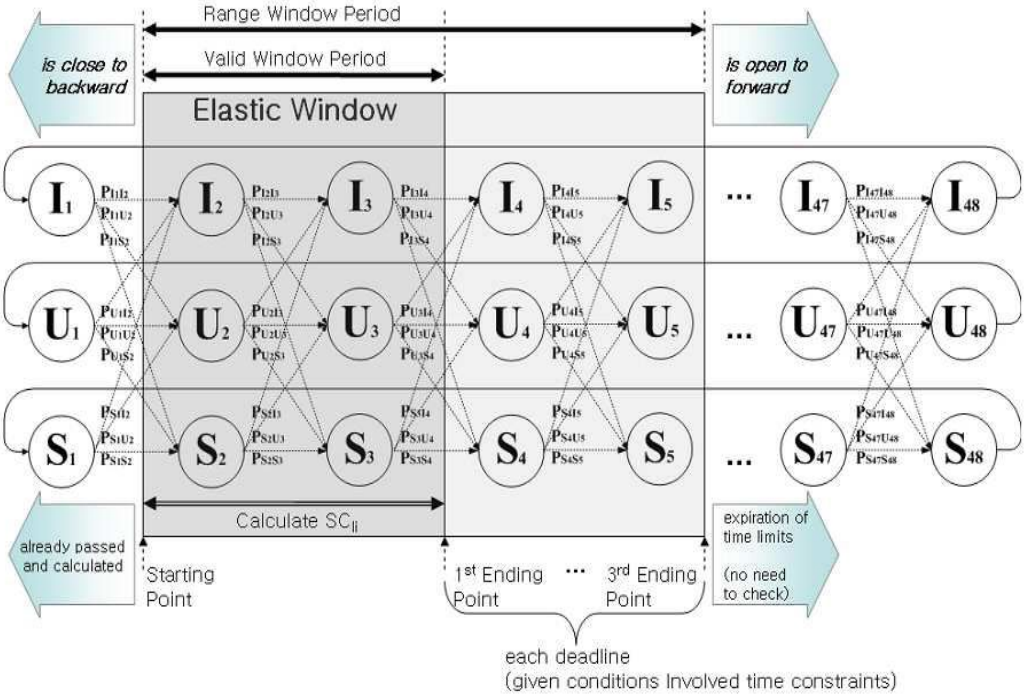
도면1



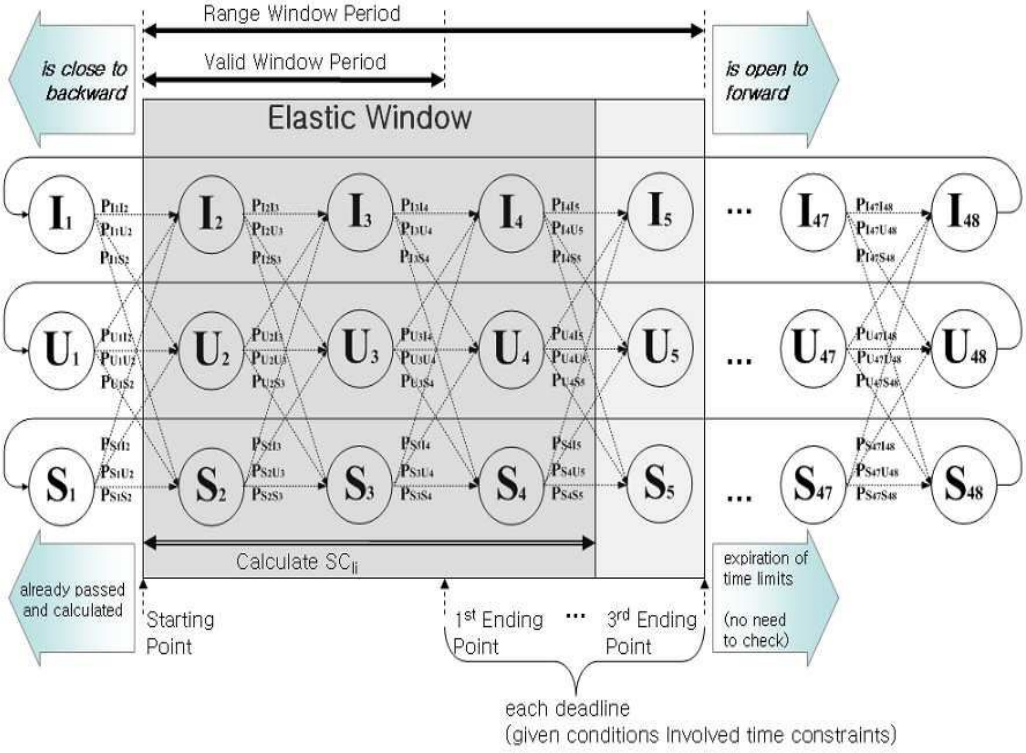
도면2



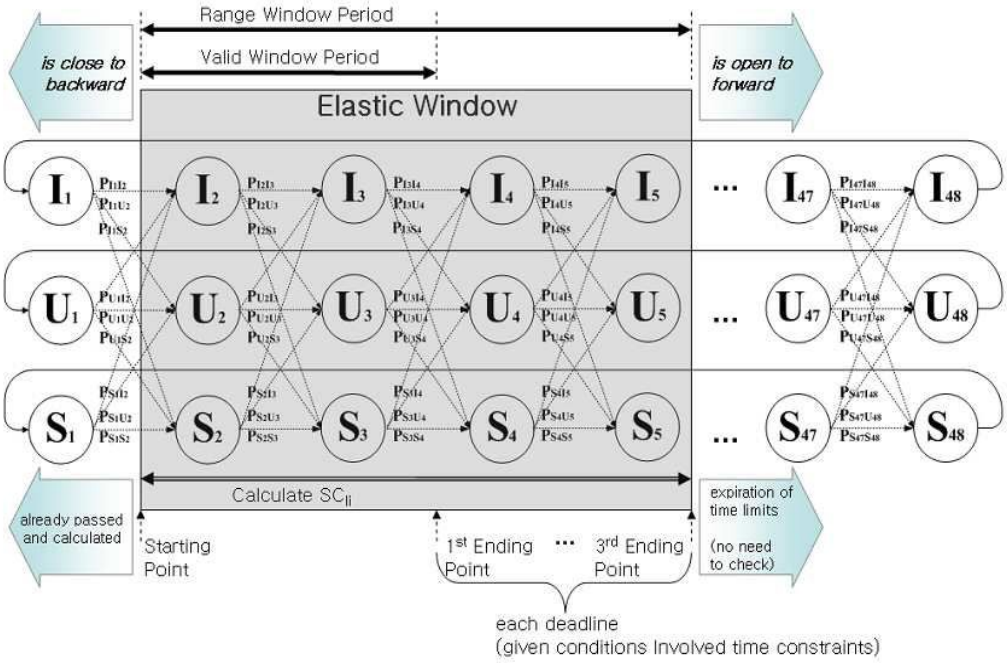
도면3


Range Window Period
Valid Window Period
Elastic Window
is close to backward
is open to forward
Calculate SC_ij
already passed and calculated
Starting Point
1st Ending Point
3rd Ending Point
expiration of time limits
(no need to check)
each deadline
(given conditions Involved time constraints)

도면4


Range Window Period
Valid Window Period
Elastic Window
is close to backward
is open to forward
already passed and calculated
Calculate SCli
Starting Point
1st Ending Point
3rd Ending Point
expiration of time limits
(no need to check)
each deadline
(given conditions involved time constraints)

도면5


Range Window Period
Valid Window Period
Elastic Window
is close to backward
is open to forward
already passed and calculated
Calculate SCli
Starting Point
1st Ending Point
3rd Ending Point
expiration of time limits
(no need to check)
each deadline
(given conditions involved time constraints)

도면6

Algorithm of SCREW

```
Float SCfi, SCneed, SCmin, SCavg, SCmax
String DType

Set WinStrPnt
Calculate SCfi, SCneed
Set EndPnt
SCneed = Get (SCmin, SCavg, SCmax)
Compare (SCfi, SCneed)

If (SCfi >= SCmin)
        Mark DType=SuperiorDT
Elseif (SCmin >= SCfi >= SCavg)
        Mark DType=ModerateDT
Elseif (SCavg >= SCfi >= SCmax)
        Mark DType=InferiorDT
Else (SCfi <= SCmax)
        Mark DType=IncongruenDT

Sort DType (SCfi);

While (WL ;){
        AllocateWL (WLi, SuperiorDT);
        If (SuperiorDT <= RequiredDT)
                AllocateWL (WLi, ModerateDT);
        AllocateWL      (Duplicate(WL      ;)      ,
InferiorDT);
        WLi++
}
```