



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2020-0129397
(43) 공개일자 2020년11월18일

(51) 국제특허분류(Int. Cl.)

G06F 9/455 (2018.01)

(52) CPC특허분류

G06F 9/45558 (2013.01)

G06F 2009/4557 (2019.08)

(21) 출원번호 10-2019-0053729

(22) 출원일자 2019년05월08일

심사청구일자 2019년05월08일

(71) 출원인

고려대학교 산학협력단

서울특별시 성북구 안암로 145, 고려대학교 (안암동5가)

(72) 발명자

유혁

서울특별시 용산구 한강대로 205, A-803호

이경운

서울특별시 성북구 보문로32길 86-7, 102호(동선동2가)

김수완

경기도 고양시 일산서구 강선로 70, 809동 1102호

(74) 대리인

김홍석

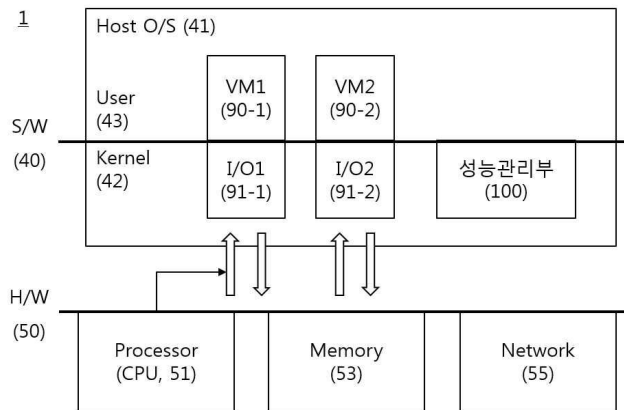
전체 청구항 수 : 총 14 항

(54) 발명의 명칭 가상화 환경에서의 성능 제어 방법 및 이를 위한 정보 처리 장치

(57) 요약

가상화 환경에서 성능 제어를 수행하는 정보 처리 장치가 개시된다. 상기 정보 처리 장치는 가상 머신과 적어도 하나의 물리 자원 간의 신호의 전달을 수행하는 입출력 스레드 및 상기 가상 머신에 대한 목표 성능 및 상기 가상 머신의 실제 성능을 비교하고, 비교 결과에 기초하여 상기 입출력 스레드에 할당되는 물리 자원을 관리하는 성능 관리부를 포함하고, 상기 성능 관리부는 상기 가상 머신의 실제 성능에 대한 정보를 획득하고, 상기 가상 머신의 실제 성능 및 상기 가상 머신에 대한 목표 성능 사이의 차이를 이용하여 성능 조절 값을 결정하고, 상기 성능 조절 값에 기초하여 상기 입출력 스레드에 할당되는 자원을 관리한다.

대표도 - 도1



90: 90-1, 90-2

91: 91-1, 91-2

(52) CPC특허분류

G06F 2009/45579 (2019.08)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711081050
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기술진흥센터
연구사업명	정보통신방송연구개발사업
연구과제명	(SW 스타랩) 성능 및 보안 SLA 보장이 가능한 차세대 클라우드 인프라SW개발
기 여 율	1/2
과제수행기관명	고려대학교 산학협력단
연구기간	2019.01.01 ~ 2019.12.31

이 발명을 지원한 국가연구개발사업

과제고유번호	1711081151
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기술진흥센터
연구사업명	정보통신방송연구개발사업
연구과제명	SDN 2.0 실현을 위한 네트워크 가상화 플랫폼 핵심 기술 및 서비스 연구
기 여 율	1/2
과제수행기관명	고려대학교 산학협력단
연구기간	2019.01.01 ~ 2019.12.31

명세서

청구범위

청구항 1

가상 머신과 적어도 하나의 물리 자원 간의 신호의 전달을 수행하는 입출력 스레드; 및

상기 가상 머신에 대한 목표 성능 및 상기 가상 머신의 실제 성능을 비교하고, 비교 결과에 기초하여 상기 입출력 스레드에 할당되는 물리 자원을 관리하는 성능 관리부를 포함하고,

상기 성능 관리부는 상기 가상 머신의 실제 성능에 대한 정보를 획득하고, 상기 가상 머신의 실제 성능 및 상기 가상 머신에 대한 목표 성능 사이의 차이를 이용하여 성능 조절 값을 결정하고, 상기 성능 조절 값에 기초하여 상기 입출력 스레드에 할당되는 자원을 관리하는,

정보 처리 장치.

청구항 2

제1항에 있어서,

상기 성능 관리부는 상기 성능 조절 값에 대응하여 상기 입출력 스레드의 프로세서에 대한 목표 사용량을 결정하고 상기 프로세서에 대한 목표 사용량을 기반으로 상기 입출력 스레드에 할당되는 상기 프로세서의 자원을 관리하는,

정보 처리 장치.

청구항 3

제2항에 있어서,

상기 성능 관리부는 상기 목표 사용량을 기반으로 상기 입출력 스레드 및 상기 프로세서 중 적어도 하나에 대한 제어 신호를 생성하는,

정보 처리 장치.

청구항 4

제2항에 있어서,

상기 성능 관리부는 상기 실제 성능 및 상기 목표 성능 사이의 차이에 대한 상기 목표 성능의 비율을 기반으로 상기 성능 조절 값을 연산하고, 상기 성능 조절 값을 상기 프로세서에 대한 현재의 사용량에 부가하거나 차감하여 상기 목표 사용량을 결정하는,

정보 처리 장치.

청구항 5

제4항에 있어서,

상기 성능 관리부는 조절 상수를 더 이용하여 상기 성능 조절 값을 연산하는,

정보 처리 장치.

청구항 6

제1항에 있어서,

상기 적어도 하나의 물리 자원은 네트워크 장치 및 메모리 장치 중 적어도 하나를 포함하는 정보 처리 장치.

청구항 7

제1항에 있어서,

상기 성능 관리부는 사용자 또는 다른 프로세서로부터 상기 가상 머신에 대한 목표 성능을 입력받는,

정보 처리 장치.

청구항 8

가상화 환경에서, 정보 처리 장치에 의해 수행되는 성능 제어 방법에 있어서,

가상 머신에 대한 목표 성능 및 상기 가상 머신에 대한 실제 성능을 획득하는 단계; 및

상기 목표 성능 및 상기 실제 성능을 기반으로 입출력 스레드에 할당되는 자원을 관리하되, 상기 입출력 스레드는 상기 가상 머신과 적어도 하나의 물리 자원 간의 신호의 전달을 수행하는 단계를 포함하는 성능 제어 방법.

청구항 9

제8항에 있어서,

상기 목표 성능 및 상기 실제 성능을 기반으로 입출력 스레드에 할당되는 자원을 관리하는 단계는,

상기 가상 머신의 실제 성능에 대한 정보를 획득하는 단계;

상기 가상 머신의 실제 성능 및 상기 가상 머신에 대한 목표 성능 사이의 차이를 이용하여 성능 조절 값을 결정하는 단계; 및

상기 성능 조절 값을 기반으로 상기 입출력 스레드에 할당되는 자원을 관리하는 단계를 포함하는,

성능 제어 방법.

청구항 10

제9항에 있어서,

상기 성능 조절 값을 기반으로 상기 입출력 스레드에 할당되는 자원을 관리하는 단계는,

상기 성능 조절 값에 대응하여 상기 입출력 스레드의 프로세서에 대한 목표 사용량을 결정하는 단계; 및

상기 프로세서에 대한 목표 사용량을 기반으로 상기 입출력 스레드에 할당되는 상기 프로세서의 자원을 관리하는 단계를 포함하는,

성능 제어 방법.

청구항 11

제10항에 있어서,

상기 성능 조절 값을 기반으로 상기 입출력 스톱에 할당되는 자원을 관리하는 단계는,

상기 목표 사용량을 기반으로 상기 입출력 스톱 및 상기 프로세서 중 적어도 하나에 대한 제어 신호를 생성하는 단계를 더 포함하는,

성능 제어 방법.

청구항 12

제10항에 있어서,

상기 가상 머신의 실제 성능 및 상기 가상 머신에 대한 목표 성능 사이의 차이를 이용하여 성능 조절 값을 결정하는 단계는, 상기 실제 성능 및 상기 목표 성능 사이의 차이에 대한 상기 목표 성능의 비율을 기반으로 상기 성능 조절 값을 연산하는 단계를 포함하고,

상기 성능 조절 값에 대응하여 상기 입출력 스톱의 프로세서에 대한 목표 사용량을 결정하는 단계는, 상기 성능 조절 값을 상기 프로세서에 대한 현재의 사용량에 부가 또는 차감하여 상기 목표 사용량을 결정하는 단계를 포함하는,

성능 제어 방법.

청구항 13

제12항에 있어서,

상기 성능 조절 값을 연산하는 단계는 조절 상수를 더 이용하여 상기 성능 조절 값을 연산하는 단계를 더 포함하는,

성능 제어 방법.

청구항 14

제8항에 있어서,

상기 적어도 하나의 물리 자원은 네트워크 장치 및 메모리 장치 중 적어도 하나를 포함하는,

성능 제어 방법.

발명의 설명

기술 분야

[0001] 본 발명은 가상화 환경에서 성능 제어 방법 및 이를 위한 정보 처리 장치에 관한 것이다.

배경 기술

[0003] 가상 머신(VM, Virtual Machine, 버추얼 머신 등으로도 표기될 수 있다)은 물리적 컴퓨팅 환경(예를 들어, 중앙 처리 장치나 메모리 장치 등)을 소프트웨어적으로 구현하여 통상의 컴퓨팅 장치와 유사하게 동작할 수 있도록 구현한 것이다. 즉, 가상 머신은 물리 머신의 자원을 이용하여 구동된다. 가상 머신에는, 가상 머신의 자원이나 소프트웨어 등을 효율적으로 관리하기 위한 운영 체제(OS, operating system, 이하 게스트 운영 체제)가 설치될 수 있다. 게스트 운영 체제는 물리 머신에 직접 설치된 운영 체제(이하 호스트 운영 체제)와 유사하게 소정의 자원에 대한 인터페이스의 제공, 입력된 데이터 등의 연산 처리 또는 필요한 동작에 대한 제어 명령 생성 등과 같은 다양한 동작을 수행할 수 있다.

[0004] 하나의 물리 머신에는 복수의 가상 머신이 설치 및 구축될 수 있다. 하나의 물리 머신 내에 설치된 복수의 가상 머신은 물리 머신의 자원을 서로 공유하면서 구동된다. 가상화(virtualization)는 물리 머신의 자원 활용률 저하를 방지하면서 다수의 가상 머신이 물리 머신의 자원을 공유할 수 있는 환경을 구축 및 제공하는 기술이다. 가상화 기술에 의해 사용자는 다양한 컴퓨팅 플랫폼 및 이러한 플랫폼을 기반으로 하는 서비스를 제공받을 수 있게 된다.

[0005] 근자에는 이와 같은 가상화 기술을 이용하면 클라우드 컴퓨팅을 구현하기도 한다. 특히, 클라우드 컴퓨팅은 서버 통합(server consolidation) 기술을 이용하여 물리적 자원을 가상화하여 통합함으로써 구현될 수도 있다. 서버 통합 기술은 상대적으로 낮은 사용률의 물리적 서버 장치들에 설치된 가상 머신들을 다른 소수의 물리적 서버 장치에 집중 설치하여 여러 가상 머신들이 하나의 물리 머신에서 동작하게 한다. 이에 따라 특정 물리적 서버 장치의 사용률을 높일 수 있게 되고, 또한 사용되지 않은 다른 물리적 서버 장치에 인가되는 전력의 차단에 따라 클라우드 데이터 센터 전체의 유지 비용을 절감할 수 있게 된다. 한편, 사용자가 이와 같이 구축된 가상 머신을 이용하는 경우, 사용 목적 등에 따라 가상 머신에 대해 어느 정도의 수준을 요구할 수 있다. 클라우드 컴퓨팅의 서비스 제공자는 서비스 수준 계약(SLA, service level agreement)을 통해 사용자의 요구 수준을 제시 및 보장하려고 하고 있다. 그러나, 여전히 성능에 대한 보장은 제대로 이뤄지지 않고 있으며, 제한적인 수준에 머물고 있다.

선행기술문헌

특허문헌

- [0007] (특허문헌 0001) 대한민국 등록특허 제1613513호 (2016.04.12.)
(특허문헌 0002) 일본국 공개특허 제2018-036724호 (2018.03.08.)

발명의 내용

해결하려는 과제

- [0008] 본 발명은 가상 머신 및 물리적 장치 사이의 입출력 성능을 제어 및 관리하여 가상 머신 간의 자원 경쟁으로 인한 성능 저하를 방지할 수 있는 가상화 환경에서의 성능 제어 방법 및 이를 위한 정보 처리 장치를 제공하는 것을 해결하고자 하는 과제로 한다.

과제의 해결 수단

- [0010] 상술한 과제를 해결하기 위하여 가상 환경에서 성능 제어 방법 및 이를 위한 정보 처리 장치가 제공된다.
- [0011] 정보 처리 장치는 가상 머신, 상기 가상 머신과 적어도 하나의 물리 자원 간의 신호의 전달을 수행하는 입출력 스레드 및 상기 가상 머신에 대한 목표 성능 및 상기 가상 머신의 실제 성능을 비교하고, 비교 결과를 기반으로 상기 입출력 스레드에 할당되는 자원을 관리하는 성능 관리부를 포함할 수 있다.
- [0012] 상기 성능 관리부는, 상기 가상 머신의 실제 성능에 대한 정보를 획득하고, 상기 가상 머신의 실제 성능 및 상기 가상 머신에 대한 목표 성능 사이의 차이를 이용하여 성능 조절 값을 결정하고, 상기 성능 조절 값을 기반으로 상기 입출력 스레드에 할당되는 자원을 관리할 수 있다.
- [0013] 상기 성능 관리부는, 상기 성능 조절 값에 대응하여 상기 입출력 스레드의 프로세서에 대한 목표 사용량을 결정하고 상기 프로세서에 대한 목표 사용량을 기반으로 상기 입출력 스레드에 할당되는 상기 프로세서의 자원을 관리할 수 있다.
- [0014] 상기 성능 관리부는, 상기 목표 사용량을 기반으로 상기 입출력 스레드 및 상기 프로세서 중 적어도 하나에 대한 제어 신호를 생성할 수 있다.
- [0015] 상기 성능 관리부는, 상기 실제 성능 및 상기 목표 성능 사이의 차이에 대한 상기 목표 성능의 비율을 기반으로

상기 성능 조절 값을 연산하고, 상기 성능 조절 값을 상기 프로세서에 대한 현재의 사용량에 부가하거나 차감하여 상기 목표 사용량을 결정할 수 있다.

- [0016] 상기 성능 관리부는, 조절 상수를 더 이용하여 상기 성능 조절 값을 연산할 수 있다.
- [0017] 상기 적어도 하나의 물리 자원은, 네트워크 장치 및 메모리 장치 중 적어도 하나를 포함할 수 있다.
- [0018] 상기 성능 관리부는, 사용자 또는 다른 프로세서로부터 상기 가상 머신에 대한 목표 성능을 입력 받을 수 있다.
- [0019] 가상 환경에서 성능 제어 방법은, 가상 머신에 대한 목표 성능 및 상기 가상 머신에 대한 실제 성능을 획득하는 단계 및 상기 목표 성능 및 상기 실제 성능을 기반으로 입출력 스레드에 할당되는 자원을 관리하되, 상기 입출력 스레드는 가상 머신과 적어도 하나의 물리 자원 간의 신호의 전달을 수행하는 단계를 포함할 수 있다.
- [0020] 상기 목표 성능 및 상기 실제 성능을 기반으로 입출력 스레드에 할당되는 자원을 관리하는 단계는, 상기 가상 머신의 실제 성능에 대한 정보를 획득하는 단계, 상기 가상 머신의 실제 성능 및 상기 가상 머신에 대한 목표 성능 사이의 차이를 이용하여 성능 조절 값을 결정하는 단계 및 상기 성능 조절 값을 기반으로 상기 입출력 스레드에 할당되는 자원을 관리하는 단계를 포함할 수 있다.
- [0021] 상기 성능 조절 값을 기반으로 상기 입출력 스레드에 할당되는 자원을 관리하는 단계는, 상기 성능 조절 값에 대응하여 상기 입출력 스레드의 프로세서에 대한 목표 사용량을 결정하는 단계 및 상기 프로세서에 대한 목표 사용량을 기반으로 상기 입출력 스레드에 할당되는 상기 프로세서의 자원을 관리하는 단계를 포함할 수 있다.
- [0022] 상기 성능 조절 값을 기반으로 상기 입출력 스레드에 할당되는 자원을 관리하는 단계는, 상기 목표 사용량을 기반으로 상기 입출력 스레드 및 상기 프로세서 중 적어도 하나에 대한 제어 신호를 생성하는 단계를 더 포함할 수 있다.
- [0023] 상기 가상 머신의 실제 성능 및 상기 가상 머신에 대한 목표 성능 사이의 차이를 이용하여 성능 조절 값을 결정하는 단계는, 상기 실제 성능 및 상기 목표 성능 사이의 차이에 대한 상기 목표 성능의 비율을 기반으로 상기 성능 조절 값을 연산하는 단계를 포함하고, 상기 성능 조절 값에 대응하여 상기 입출력 스레드의 프로세서에 대한 목표 사용량을 결정하는 단계는, 상기 성능 조절 값을 상기 프로세서에 대한 현재의 사용량에 부가 또는 차감하여 상기 목표 사용량을 결정하는 단계를 포함할 수 있다.
- [0024] 상기 성능 조절 값을 연산하는 단계는, 조절 상수를 더 이용하여 상기 성능 조절 값을 연산하는 단계를 더 포함할 수 있다.
- [0025] 상기 적어도 하나의 물리 자원은, 네트워크 장치 및 메모리 장치 중 적어도 하나를 포함할 수 있다.
- [0026] 정보 처리 장치는, 복수의 가상 머신, 상기 복수의 가상 머신 각각에 대응하고, 상기 복수의 가상 머신과 적어도 하나의 물리 자원 간의 신호의 전달을 수행하는 복수의 입출력 스레드 및 상기 복수의 가상 머신의 요구 성능에 따라서 상기 복수의 입출력 스레드 각각에 대해 자원을 할당하는 성능 관리부를 포함할 수 있다.

발명의 효과

- [0028] 상술한 가상화 환경에서 성능 제어 방법 및 이를 위한 정보 처리 장치에 의하면, 가상 머신 및 물리적 장치, 예컨대 네트워크 장치나 메모리 장치 사이의 입출력 성능을 제어 및 관리하여 가상 머신 간의 자원 경쟁으로 인한 성능 저하를 방지할 수 있는 효과를 얻을 수 있다.
- [0029] 또한, 중앙 처리 장치 등과 같은 프로세서의 자원을 효율적으로 사용할 수 있게 되고, 이에 따라 프로세서의 가용 자원의 추가적인 확보가 가능해지는 효과도 얻을 수 있다.
- [0030] 또한, 클라우드 컴퓨팅의 서비스 제공자가 사용자의 요구 수준에 부합하는 성능을 적절하게 보장할 수 있게 되는 효과도 얻을 수 있다.
- [0031] 또한, 별도의 추가적인 하드웨어 장비의 설치나 교체 없이 가상 머신의 동작의 효율화 및 성능의 안정화를 이룰 수 있어 서비스 품질의 향상의 효과와 더불어 비용 절감의 효과도 얻을 수 있다.

도면의 간단한 설명

- [0033] 본 발명의 상세한 설명에서 인용되는 도면을 보다 충분히 이해하기 위하여 각 도면의 상세한 설명이 제공된다.
- 도 1은 정보 처리 장치의 일 실시예를 설명하기 위한 도면이다.
- 도 2는 도 1에 도시된 성능 관리부의 일 실시예를 설명하기 위한 블록도이다.
- 도 3은 도 2에 도시된 성능 감시부 및 성능 제어부의 일 실시예를 설명하기 위한 블록도이다.
- 도 4는 도 1에 도시된 성능 관리부의 적용 전 및 적용 후의 동작을 설명하기 위한 제1 도이다.
- 도 5는 도 1에 도시된 성능 관리부의 적용 전 및 적용 후의 동작을 설명하기 위한 제2 도이다.
- 도 6은 가상화 환경에서 도 1에 도시된 정보 처리 장치에 의해 수행되는 성능 제어 방법을 설명하기 위한 흐름도이다.

발명을 실시하기 위한 구체적인 내용

- [0034] 이하 전반적으로 동일 참조 부호는 특별한 사정이 없는 한 동일 구성요소를 지칭한다. 또한, '부'가 부가된 용어는, 소프트웨어적 및/또는 하드웨어적 부품으로 구현될 수 있으며, 실시예에 따라 '부'가 하나의 소프트웨어적 및/또는 하드웨어적 부품으로 구현되거나, 하나의 '부'가 복수의 소프트웨어적 및/또는 하드웨어적 부품들로 구현되는 것도 가능하다.
- [0035] 어떤 부분이 다른 부분과 연결되어 있다고 할 때, 이는 어떤 부분과 다른 부분에 따라서 물리적 연결을 의미할 수도 있고, 또는 전기적으로 연결된 것을 의미할 수도 있다. 또한, 어떤 부분이 다른 부분을 포함한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 부분 이외의 또 다른 부분을 제외하는 것이 아니며, 설계자의 선택에 따라서 또 다른 부분을 더 포함할 수 있음을 의미한다.
- [0036] '제1' 이나 '제2' 등의 용어는 하나의 부분을 다른 부분으로부터 구별하기 위해 사용되는 것으로, 특별한 기재가 없는 이상 이들이 순차적인 표현을 의미하는 것은 아니다. 또한, 단수의 표현은 문맥상 명백하게 예외가 있지 않는 한, 복수의 표현을 포함할 수 있다.
- [0037] 이하, 도 1 내지 도 6을 참조하여 가상화 환경에서의 성능 제어를 수행할 수 있는 정보 처리 장치의 일 실시예에 대해서 설명하도록 한다.
- [0039] 도 1은 정보 처리 장치의 일 실시예를 설명하기 위한 도면이다. 도 1을 참조하면, 정보 처리 장치(1)는 하드웨어(50) 및 하드웨어(50)를 기반으로 구동하는 소프트웨어(40)를 포함할 수 있다.
- [0040] 하드웨어(50)는 정보 처리 장치(1)에 장착되어 정보 처리 장치(1)의 동작을 수행하거나 동작을 지원하는 물리적 장치를 의미하며, 소프트웨어(40, 예컨대 호스트 운영 체제(41) 등)에 필요한 자원을 제공하여 소프트웨어(40)가 소정의 동작을 수행하도록 한다. 하드웨어(50)는 프로세서(51), 메모리 장치(53), 및 네트워크 장치(55) 등을 포함할 수 있다.
- [0041] 프로세서(51)는 정보 처리 장치(1)를 전반적으로 또는 부분적으로 관리 및 제어하고 정보 처리 장치(1) 또는 이에 설치된 소프트웨어(프로그램 등)의 동작에 필요한 연산 처리 등을 수행할 수 있다. 프로세서(51)는 호스트 운영 체제(41) 및/또는 호스트 운영 체제(41)에 설치된 적어도 하나의 가상 머신(90: 90-1, 90-2)의 구동에 필요한 자원을 할당하여 제공할 수 있다. 또한, 프로세서(51)는 호스트 운영 체제(41)의 커널 영역(42)에서 동작하는 적어도 하나의 입출력 스레드(91: 91-1, 91-2)에 필요한 자원을 할당하여 제공할 수도 있다.
- [0042] 프로세서(51)는 중앙 처리 장치(CPU, Central Processing Unit), 마이크로 컨트롤러 유닛(MCU, Micro Controller Unit), 마이컴(Micom, Micro Processor), 애플리케이션 프로세서(AP, Application Processor), 전자 제어 유닛(ECU, Electronic Controlling Unit) 및/또는 각종 연산 처리 및 제어 신호의 생성이 가능한 다른 전자 장치 등을 포함할 수 있다. 이들 장치는 하나 또는 둘 이상의 반도체 칩 및 관련 부품을 이용하여 구현 가능하다.
- [0043] 메모리 장치(53)는 데이터를 기록 및 저장할 수 있도록 마련된다. 메모리 장치(53)는 프로세서(51), 운영 체제(41) 및/또는 운영 체제(41) 내에서 동작하는 여러 프로그램이나 가상 머신(90: 90-1, 90-2)으로부터 데이터를 수신하여 기록하고, 기록된 데이터를 이들 장치(41, 90, 51 등)에 제공할 수도 있다.
- [0044] 메모리 장치(53)와 가상 머신(90: 90-1, 90-2)이 요청, 명령(예를 들어, 입출력 처리 요청 등) 또는 데이터를

상호 송수신하는 경우, 가상 머신(90: 90-1, 90-2) 각각에 대응하는 입출력 스레드(91: 91-1, 91-2)가 이용될 수 있다. 즉, 각 가상 머신(90: 90-1, 90-2)의 데이터는 입출력 스레드(91: 91-1, 91-2)를 통하여 메모리 장치(53)로 전달될 수 있다. 따라서, 입출력 스레드(91: 91-1, 91-2)에 충분한 자원(예를 들어, 중앙 처리 장치 자원)이 할당되지 않은 경우, 가상 머신(90: 90-1, 90-2)에 충분한 자원이 할당되어 있다고 하더라도, 가상 머신(90: 90-1, 90-2)과 메모리 장치(53) 사이의 데이터 송수신은 지연되기 쉽다.

[0045] 메모리 장치(53)는 주기억장치 및 보조기억장치 중 적어도 하나를 포함할 수 있다. 주기억장치는 롬(ROM) 및/또는 램(RAM)을 이용하여 구현될 수 있으며, 롬은 피롬(PROM), 이프롬(EPROM), 이이프롬(EEPROM) 및/또는 마스크 롬(MASK-ROM) 등을 포함하고, 램은 디램(DRAM) 및/또는 에스램(SRAM) 등을 포함한다. 보조기억장치는 플래시 메모리 장치, SD(Secure Digital) 카드, 솔리드 스테이트 드라이브(SSD, Solid State Drive), 하드 디스크 드라이브(HDD, Hard Disc Drive), 자기 드럼, 콤팩트 디스크(CD), 디브이디(DVD) 또는 레이저 디스크 등과 같은 광 기록 매체(optical media), 자기 테이프, 광자기 디스크 및/또는 플로피 디스크 등과 같이 데이터를 영구적 또는 반영구적으로 저장 가능한 적어도 하나의 저장 매체를 이용하여 구현될 수 있다.

[0046] 네트워크 장치(55)는 다른 정보 처리 장치, 외장 메모리 장치 및/또는 이외 정보 처리 장치(1)와 직간접적으로 연결 가능한 다양한 장치와 통신을 수행할 수 있도록 마련된다. 네트워크 장치(55)는 통신 칩, 통신 안테나, 통신 단자 및/또는 이외 통신 부품(예를 들어, 증폭기 등) 등과 같이 소정의 부품들의 집합하여 구현될 수 있다.

[0047] 일 실시예에 의하면, 네트워크 장치(55)는 입출력 스레드(91: 91-1, 91-2)를 이용하여 각각의 입출력 스레드(91: 91-1, 91-2)에 대응하는 가상 머신(90: 90-1, 90-2)과 요청이나 명령(예를 들어, 입출력 처리 요청 등)이나 데이터를 송수신할 수 있다. 따라서, 상술한 바와 같이 각각의 입출력 스레드(91: 91-1, 91-2)에 적절한 자원이 할당되지 않았다면, 네트워크 장치(55) 및 가상 머신(90: 90-1, 90-2) 사이의 작업 처리 역시 지연되기 쉽다.

[0048] 네트워크 장치(55)는, 예를 들어, 유선 통신 네트워크, 무선 통신 네트워크 또는 이들의 조합에 접속할 수 있다. 여기서, 유선 통신 네트워크는, 케이블을 이용하여 구축된 네트워크를 의미하며, 케이블은, 예를 들어, 페어 케이블, 동축 케이블, 광섬유 케이블 또는 이더넷 케이블 등을 포함할 수 있다. 무선 통신 네트워크는 근거리 통신 네트워크 및 원거리 통신 네트워크 중 적어도 하나를 이용하여 구축된 네트워크를 의미한다. 근거리 통신 네트워크는, 예를 들어, 와이 파이(Wi-Fi), 와이파이 다이렉트(Wi-Fi Direct), 블루투스(Bluetooth), 저전력 블루투스(Bluetooth Low Energy), 지그비(zigbee), 엔에프씨(NFC, Near Field Communication) 및/또는 캔(CAN) 통신 등을 포함할 수 있다. 원거리 통신 네트워크는, 유선 통신 네트워크, 예를 들어, 3GPP, 3GPP2 또는 와이맥스 계열 등의 이동 통신 표준을 기반으로 구현된 것일 수 있다.

[0049] 이외에도 정보 처리 장치(1)는, 예를 들어, 키보드 장치나 마우스 장치 등과 같은 입력 인터페이스 장치나, 모니터 장치나 스피커 장치 등과 같은 출력 인터페이스 장치 등을 더 포함할 수도 있다.

[0050] 상술한 정보 처리 장치(1)는 가상 머신(90-1, 90-2)을 설치 가능한 적어도 하나의 장치로 구현될 수 있다. 예를 들어, 정보 처리 장치(1)는 데스크톱 컴퓨터, 랩톱 컴퓨터, 서버용 하드웨어 장치, 네트워크 결합 스토리지 장치(NAS), 스마트 폰, 태블릿 피씨, 두부 장착형 디스플레이(HMD, Head Mounted Display) 장치, 스마트 시계, 디지털 텔레비전, 셋톱 박스, 내비게이션 장치, 개인용 디지털 보조기(PDA, Personal Digital Assistant), 가전 기기, 기계 장치, 로봇, 차량 제어 장치, 휴대용 게임기, 전자 칠판, 전자 광고판, 음향 재생 장치 및/또는 이외 가상 머신을 설치 가능한 다양한 정보 처리 장치 중 적어도 하나 등을 포함할 수 있다.

[0051] 소프트웨어(40)는 정보 처리 장치(1)에 의해 수행되는 대부분의 프로세스의 관리, 처리 및/또는 제어 등을 수행하기 위한 운영 체제(41)를 포함할 수 있다. 운영 체제(41)는, 예를 들어, 마이크로소프트 윈도우즈, 맥 오에스(Mac OS), 리눅스, 유닉스 또는 유닉스에서 파생된 각종 운영 체제를 포함할 수 있다. 운영 체제(41)에는 적어도 하나의 가상 머신(90: 90-1, 90-2)이 설치될 수 있으며, 운영 체제(41)는, 적어도 하나의 가상 머신(90: 90-1, 90-2)에 대응하여, 호스트 운영 체제라고 표현될 수도 있다.

[0052] 호스트 운영 체제(41)는 사용자 영역(43) 및 커널 영역(42)을 포함할 수 있다. 사용자 영역(43)에서는 정보 처리 장치(1)가 구동 가능한 다양한 응용 프로그램(애플리케이션이나 앱 등으로 표현 가능함)이 실행될 수 있다. 예를 들어, 각각의 가상 머신(90: 90-1, 90-2)은 사용자 영역(43)에서 구동하여 사용자에게 여러 서비스를 제공할 수 있다.

[0053] 가상 머신(90: 90-1, 90-2)은 물리적 장치(일례로, 정보 처리 장치(1) 자체 또는 정보 처리 장치(1)의 물리적 부품(51, 53, 55 등) 등을 가상으로 구현한 소프트웨어를 의미한다. 가상 머신(90: 90-1, 90-2)에는 사용자 또

는 설계자의 선택에 따라 소정의 운영 체제(이하 게스트 운영 체제)가 설치될 수 있다. 이 경우, 게스트 운영 체제는 호스트 운영 체제(41)와 동일한 운영 체제일 수도 있고, 또는 상이한 운영 체제일 수도 있다. 예를 들어, 호스트 운영 체제(41)는 리눅스 운영 체제이고, 게스트 운영 체제는 마이크로소프트 윈도우즈 운영 체제일 수도 있다. 하나의 호스트 운영 체제(41)에는 하나 또는 복수의 가상 머신(90: 90-1, 90-2)이 설치될 수 있다. 복수의 가상 머신(90: 90-1, 90-2)이 설치된 경우, 각각의 가상 머신(90: 90-1, 90-2)은 상호 물리적으로 분리된 장치나 운영 체제처럼 동작할 수 있다. 이 경우, 복수의 가상 머신(90: 90-1, 90-2) 중 적어도 둘에는 서로 동일한 운영 체제가 설치될 수도 있고 또는 서로 상이한 운영 체제가 설치될 수도 있다.

[0054] 일 실시예에 의하면, 가상 머신(90: 90-1, 90-2)은 커널 기반 가상 머신(KVM, Kernel based Virtual Machine)을 포함할 수 있다. 커널 기반 가상 머신은 리눅스 기반의 가상화 플랫폼으로, 리눅스를 호스트 운영 체제(41)로 이용하여 커널 모듈 형태로 가상화 환경을 제공할 수 있다.

[0055] 커널(kernel)은 정보 처리 장치(1)의 시스템을 전반적으로 관리 및 통제하는 소프트웨어를 의미한다. 예를 들어, 커널은 프로세스 관리, 자원 관리 또는 보안 등 다양한 동작을 수행할 수 있게 설계된다. 이와 같은 커널은, 예를 들어, 단일형 커널, 마이크로 커널, 하이브리드 커널, 엑소커널(exokernel) 또는 나노커널 등을 포함할 수 있다. 커널에 의해 구현되는 커널 영역(42)에서는 적어도 하나의 커널 스레드(kernel thread)가 구동될 수 있다. 커널 스레드는 적어도 하나의 입출력 스레드(91: 91-1, 91-2, I/O thread)를 포함할 수 있다.

[0056] 입출력 스레드(91: 91-1, 91-2)는 가상 머신(90: 90-1, 90-2)이 생성한 요청이나 데이터 등을 물리적 장치(51, 53, 55 등)로 전달하거나 또는 물리적 장치(51, 53, 55 등)의 요청이나 데이터 등을 가상 머신(90: 90-1, 90-2)으로 전달할 수 있다. 입출력 스레드(91: 91-1, 91-2)는 가상 머신(90: 90-1, 90-2)마다 각각의 가상 머신(90: 90-1, 90-2)에 대응하여 마련될 수 있다. 예를 들어, 제1 가상 머신(90-1)에는 제1 입출력 스레드(91-1)가 대응하여 마련되고, 제2 가상 머신(90-2)에는 제2 입출력 스레드(91-2)가 대응하여 마련될 수 있다. 실시예에 따라서, 다수의 가상 머신(90)에 하나의 입출력 스레드(91)가 마련될 수도 있고, 하나의 가상 머신(90)에 다수의 입출력 스레드(91)가 마련되는 것도 가능하다. 입출력 스레드(91: 91-1, 91-2)는 상술한 커널 기반 가상 머신 환경 하에서 동작할 수도 있으나, 이에 한정되는 것은 아니다.

[0057] 일 실시예에 의하면, 정보 처리 장치(1)는 성능 관리부(100)를 포함할 수 있다. 또한, 필요에 따라 정보 처리 장치(1)는 자원 관리부(도 2의 60 참조)를 더 포함하는 것도 가능하다.

[0058] 성능 관리부(100)는 하나 또는 둘 이상의 가상 머신(90: 90-1, 90-2)의 성능을 관리하거나 및/또는 이에 필요한 여러 제어 동작을 수행할 수 있도록 설계된 것일 수 있다. 만약 복수의 가상 머신(90: 90-1, 90-2)이 호스트 운영 체제(41)에 설치된 경우, 성능 관리부(100)는 복수의 가상 머신(90: 90-1, 90-2)을 각각 관리 및 제어할 수도 있다. 성능 관리부(100)는 도 1에 도시된 바와 같이, 커널 영역(42)에서 동작할 수 있으며, 예를 들어, 가상화 환경에서 동작하는 커널 모듈일 수 있다. 커널 모듈이란, 필요 시 커널 영역(42)에 로드되거나 또는 언로드되어 커널 영역(42)에서 실행되고 종료되는 코드(프로그램)를 의미할 수 있다. 실시예에 따라서, 성능 관리부(100)는 커널 영역(42) 이외의 영역에서 동작하도록 설계되는 것도 가능하다. 예를 들어, 성능 관리부(100)는 사용자 영역(41) 내에서 소정의 동작을 수행하도록 설계될 수도 있고 또는 소정의 프로세서(51)에 미리 임베디드되어 소정의 동작을 수행하도록 설계될 수도 있다.

[0059] 일 실시예에 의하면, 성능 관리부(100)는 입출력 스레드(91: 91-1, 91-2)가 자원을 점유하는 정도나 비중(사용량, 점유량 또는 점유율 등으로 표현 가능함)을 관리할 수 있다. 예를 들어, 성능 관리부(100)는 입출력 스레드(91: 91-1, 91-2) 각각에 제공되는 프로세서(51, 일례로 중앙 처리 장치)의 자원을 분배하여 할당함으로써, 프로세서(51)의 자원에 대한 적어도 하나의 입출력 스레드(91: 91-1, 91-2) 각각의 사용량을 관리, 조절 또는 제어할 수 있다. 이에 따라 입출력 스레드(91: 91-1, 91-2)의 자원에 대한 사용량을 관리하기 위한 스케줄링 처리가 가능해진다. 프로세서(51)의 자원에 대한 사용량은 중앙 처리 장치의 대역폭 등으로 표현될 수도 있다. 여기서, 중앙 처리 장치의 대역폭은 중앙 처리 장치가 제공하는 자원의 사용량을 나타내는 값으로 퍼센티지로 표현될 수 있으며, 만약 하나의 물리 코어를 완전히 사용하고 있는 경우에는 중앙 처리 장치의 대역폭은 100%로 표현될 수 있다.

[0060] 성능 관리부(100)는 커널 기반 가상 머신 환경에서 동작하도록 마련된 것일 수도 있다. 이 경우, 만약 성능 관리부(100)가 동작하지 않는 경우, 정보 처리 장치(1)는 통상적인 커널 기반 가상 머신 환경과 동일하게 동작하게 되고, 반대로 만약 성능 관리부(100)가 동작하는 경우에는 커널 기반 가상 머신 환경 하에서 입출력 스레드(91: 91-1, 91-2)의 성능을 제어 가능하게 되고, 이에 따라 물리적 장치, 일례로 메모리 장치(53)나 네트워크 장치(55) 등에 대한 입출력 성능의 관리 및 제어가 가능해진다.

- [0061] 성능 관리부(100)는 애플리케이션의 형태로 구현될 수도 있다. 이 경우 성능 관리부(100)는 설계자 등에 의해 커널 영역(42)에 직접 설치된 애플리케이션일 수도 있고, 또한 유선 또는 무선 통신 네트워크를 통해 접속 가능한 전자 소프트웨어 유통망을 통하여 획득된 애플리케이션일 수도 있다.
- [0063] 도 2는 도 1에 도시된 성능 관리부의 일 실시예를 설명하기 위한 블록도이고, 도 3은 도 2에 도시된 성능 감시부 및 성능 제어부의 일 실시예를 설명하기 위한 블록도이다.
- [0064] 도 2에 도시된 일 실시예에 의하면, 성능 관리부(100)는 목표 성능 수신부(110), 성능 감시부(130) 및 성능 제어부(150)를 포함할 수 있다.
- [0065] 목표 성능 수신부(110)는 각각의 가상 머신(90-1 내지 90-N, 여기서 N은 1 이상의 자연수)에 대해 요구되는 성능, 각각의 가상 머신(90-1 내지 90-N)의 특정한 자원에 대해 요구되는 성능 및/또는 각각의 가상 머신(90-1 내지 90-N)이 요구하는 성능(이하 목표 성능)을 수신하여 설정할 수 있다. 여기서, 가상 머신(90-1 내지 90-N)의 특정한 자원에 대해 요구되는 성능은, 예를 들어, 메모리 장치(53)에 대한 입출력 성능을 포함할 수도 있고 및/또는 네트워크 장치(55)에 대한 입출력 성능을 포함할 수도 있다. 이 경우, 메모리 장치(53)의 입출력에 대한 목표 성능은 대역폭(MB/s)을 이용하여 입력될 수 있으며, 네트워크 장치(55)의 입출력에 대한 목표 성능은 초당 처리되는 패킷의 수나 대역폭(Mbps) 등을 이용하여 입력될 수 있다.
- [0066] 목표 성능 수신부(110)는 입력 인터페이스(미도시) 또는 프로세서(51) 등으로부터 목표 성능을 입력받을 수 있다. 예를 들어, 사용자의 조작에 따라서 입력 인터페이스는 전기적 신호를 출력하고, 목표 성능 수신부(110)는 입력 인터페이스가 출력한 전기적 신호 또는 이에 대응하는 데이터를 수신함으로써 목표 성능을 수신할 수 있다. 목표 성능 수신부(110)는 수신한 목표 성능을 성능 제어부(150)로 전달할 수 있다.
- [0067] 성능 감시부(130)는 가상 머신(90-1 내지 90-N) 또는 가상 머신(90-1 내지 90-N) 각각에 대응하는 입출력 스레드(91-1 내지 91-N)에 대한 정보를 수집할 수도 있다. 예를 들어, 성능 감시부(130)는 가상 머신(90-1 내지 90-N) 또는 입출력 스레드(91-1 내지 91-N)가 실제로 사용하고 있는 자원의 사용량(실제 자원 사용량)에 대한 정보를 수집 및 획득하여 이들(90-1 내지 90-N, 91-1 내지 91-N)의 동작을 감시할 수 있다.
- [0068] 또한, 성능 감시부(130)는 가상 머신(90-1 내지 90-N) 또는 입출력 스레드(91-1 내지 91-N)의 실제 성능(즉, 현재의 성능)을 판단하고, 판단된 실제 성능을 기반으로 성능 제어가 필요한지 여부 등을 판단할 수도 있다. 이를 위해서 성능 감시부(130)는 수집한 정보를 기반으로 가상 머신(90-1 내지 90-N) 또는 입출력 스레드(91-1 내지 91-N)에 대한 성능 정보를 획득할 수도 있다.
- [0069] 일 실시예에 의하면, 도 3에 도시된 바와 같이 성능 감시부(130)는, 실제 성능 결정 및 제어 필요 여부 판단을 위해서, 제어 목표 선택부(131), 성능 정보 획득부(133) 및 목표 달성 판단부(135)를 포함할 수 있다.
- [0070] 제어 목표 선택부(131)는 복수의 가상 머신(90-1 내지 90-N) 각각에 대응하는 복수의 입출력 스레드(91-1 내지 91-N) 중에서 제어 대상이 될 적어도 하나의 입출력 스레드, 예를 들어 제1 입출력 스레드(91-1)를 선택할 수 있다. 실시예에 따라 복수의 입출력 스레드(91-1 내지 91-N)가 존재하는 경우, 제어 목표 선택부(131)는 복수의 입출력 스레드(91-1 내지 91-N) 모두를 각각 제어 목표로 선택할 수도 있고 또는 복수의 입출력 스레드(91-1 내지 91-N) 중 일부의 입출력 스레드(91-1 내지 91-N 중 일부)만을 제어 목표로 선택할 수도 있다. 또한, 제어 목표 선택부(131)는 복수의 입출력 스레드(91-1 내지 91-N)의 전부 또는 일부를 순차적으로 또는 임의적으로 제어 목표로 선택할 수도 있다.
- [0071] 성능 정보 획득부(133)는 가상 머신(90-1 내지 90-N) 또는 입출력 스레드(91-1 내지 91-N)의 성능 연산을 위해 필요한 정보를 획득하고 이를 기반으로 이들 가상 머신(90-1 내지 90-N) 또는 입출력 스레드(91-1 내지 91-N)의 실제 성능을 결정할 수도 있다. 이 경우, 성능 정보 획득부(133)는 상술한 제어 목표 선택부(131)에 의해 선택된 제1 입출력 스레드(91-1) 또는 제1 입출력 스레드(91-1)에 대응되는 가상 머신, 즉 제1 가상 머신(90-1)에 대한 필요한 정보를 수집하거나 또는 이들의 실제 성능을 결정할 수 있다. 여기서, 성능 연산을 위해 필요한 정보는, 예를 들어, 실제 네트워크 장치(55)의 입출력 동작 시의 초당 패킷 수나 대역폭 또는 메모리 장치(53)의 입출력 동작 시의 대역폭 등을 포함할 수 있다. 성능 정보 획득부(133)는 성능 정보의 획득을 위해서 별도로 마련된 자원 관리부(60) 및/또는 장치(1)에 대한 소정의 통계 정보를 이용하여 실시간 자원 사용량 및 실제 성능을 계산할 수도 있다.
- [0072] 목표 달성 판단부(135)는 목표 성능 수신부(110)로부터 목표 성능을 수신하고, 성능 정보 획득부(133)가 획득한

실제 성능을 목표 성능과 비교하여 목표 달성 여부를 판단할 수 있다. 구체적으로 예를 들면, 목표 달성 판단부(135)는 실제 성능과 목표 성능 사이의 차이를 연산하고 실제 성능과 목표 성능의 차이가 미리 정의된 값과 동일하거나 또는 일정한 범위(예를 들어, 오차 범위) 내의 값인 경우에는 목표가 달성되었다고 판단할 수 있다. 여기서 미리 정의된 값은 0을 포함할 수 있다. 다시 말해서, 목표 달성 판단부(135)는 목표 성능과 실제 성능의 차이가 0이거나(또는 목표 성능과 실제 성능이 동일하거나) 또는 0을 기준으로 미리 정의된 범위 이내의 값인 경우(목표 성능-실제 성능=0 or 0+a, a는 임의의 실수), 가상 머신(90-1 내지 90-N) 또는 입출력 스레드(91-1 내지 91-N)가 목표 성능에 도달했다고 판단할 수 있다. 여기서 미리 정의된 범위 이내의 값은, 예를 들어, 0에 근사한 값을 포함할 수 있다. 반대로 실제 성능과 목표 성능 사이가 0이 아니거나 또는 미리 정의된 범위 외의 값을 갖는 경우(즉, 목표 성능-실제 성능 ≠ 0 or 0+a), 목표 달성 판단부(135)는 목표가 달성되지 않았다고 판단할 수 있다.

[0073] 성능 감시부(130)의 판단 결과는 성능 제어부(150)로 전달될 수 있으며, 필요에 따라서 성능 감시부(130)에 의해 획득된 정보도 성능 제어부(150)로 전달될 수 있다. 예를 들어, 목표가 달성되지 않았다고 판단되면, 목표 달성 판단부(135)는 목표가 달성되지 않았다는 판단 결과를 성능 제어부(150)로 전달할 수 있으며, 판단 결과와 더불어 목표 성능, 실제 성능 및 실제 자원 사용량에 대한 정보도 성능 제어부(150)로 전달할 수 있다. 필요에 따라서, 목표 달성 판단부(135)에 의해 연산된 목표 성능 및 실제 성능의 차이 역시 성능 제어부(150)로 전달될 수도 있다. 반대로 만약 목표가 달성되었다고 판단되면, 목표 달성 판단부(135)는 성능 제어부(150)로 어떠한 판단 결과도 전달하지 않을 수도 있고, 실시예에 따라서, 목표 달성에 대한 정보를 성능 제어부(150)로 전송할 수도 있다.

[0074] 성능 감시부(130)에 의한 실제 자원 사용량의 수집 또는 실제 성능의 획득은 주기적으로 수행될 수도 있고 또는 비주기적으로 수행될 수도 있다. 예를 들어, 성능 감시부(130)는 미리 정의된 주기(예를 들어, 1초)마다 적어도 하나의 입출력 스레드(91-1 내지 91-N), 예를 들어 선택된 제1 입출력 스레드(91-1)의 자원 사용량을 수집할 수도 있다. 여기서, 제1 입출력 스레드(91-1)의 자원 사용량은 메모리 장치(53)와의 통신을 수행하기 위한 프로세서(51)에 대한 제1 입출력 스레드(91-1)의 사용량 및 네트워크 장치(55)와의 통신을 수행하기 위한 프로세서(51)에 대한 제1 입출력 스레드(91-1)의 자원 사용량 중 적어도 하나를 포함할 수도 있다.

[0075] 성능 제어부(150)는 성능 감시부(130)가 도출한 실제 성능 및 목표 달성 판단 결과를 기반으로, 가상 머신(90-1 내지 90-N) 또는 입출력 스레드(91-1 내지 91-N) 각각에 대한 성능 조절 값을 결정하고, 성능 조절 값을 기반으로 입출력 스레드(91-1 내지 91-N)의 실제 사용량이 목표 사용량과 동일 또는 근사해질 수 있도록 제어할 수 있다.

[0076] 일 실시예에 의하면, 성능 제어부(150)는 목표 사용량 결정부(151) 및 제어 신호 생성부(153)를 포함할 수 있다.

[0077] 목표 사용량 결정부(151)는 성능 감시부(130)로부터 신호의 수신에 응하여 목표 사용량을 결정함으로써 자원 스케줄링 조절을 수행할 수 있다. 다시 말해서, 목표 사용량 결정부(151)는, 제어 목표로 선택된 제1 입출력 스레드(91-1)의 실제 성능이나 이에 대응하는 제1 가상 머신(90-1)의 실제 성능이 사용자 등이 요구하는 목표 성능에 미치지 못하는 경우에, 제1 입출력 스레드(91-1) 또는 제1 가상 머신(90-1)이 목표 성능을 달성할 수 있도록 소정의 자원에 대한 제1 스레드(91-1)의 목표 사용량을 결정할 수 있다. 이 경우, 목표 사용량 결정부(151)는 실제 성능, 목표 성능 및 실제 사용량을 기반으로 제1 스레드(91-1)의 목표 사용량을 결정할 수 있다.

[0078] 구체적으로 예를 들어, 목표 사용량 결정부(151)는 하기의 수학식 1 및 수학식 2를 기반으로 목표 사용량(즉, 자원의 사용량)을 결정할 수 있다.

수학식 1

$$C_{i+1} = C_i + k \cdot \frac{P_g - P_r}{P_g}$$

[0079]

[0080] 여기서 C_i 는 특정한 시점(i)에서의 자원(예를 들어, 중앙 처리 장치)의 실제 사용량을 의미하고, C_{i+1} 은 다음 시점(i+1)에서의 자원에 대한 목표 사용량을 의미할 수 있다. 또한, k는 목표 사용량의 변화 정도를 조절하기 위한 값(이하 조절 상수)로 0 및 1 사이의 값 중 어느 하나의 실수 값을 갖는다. 만약 조절 상수(k)가 상대적으로 0에 근사하다면 목표 사용량은 보다 세밀하게 조절되며, 반대로 조절 상수(k)가 상대적으로 1에 근사하다면

목표 사용량은 보다 큰 폭으로 조절되게 된다. 조절 상수(k)는 0.1로 주어질 수 있으나, 이는 예시적인 것이며 설계자나 사용자의 선택에 따라 조절 상수(k)는 다양하게 정의 가능하다. P_g 는 수치화된 목표 성능을 의미하고, P_r 는 수치화된 실제 성능을 의미한다. 목표 성능(P_g)은 목표 성능 수신부(110)에 의해 획득되고 실제 성능(P_r)

은 성능 정보 획득부(133)에 의해 획득된 것일 수 있다. 수학식 1의 우변의 $\frac{P_g - P_r}{P_g}$ 은 성능 조절 값을 의미할 수 있다.

[0081] 상술한 수학식 1은 하기의 수학식 2와 같이 다시 쓰일 수 있다.

수학식 2

$$Q_{i+1} = Q_i + F \cdot k \cdot \frac{P_g - P_r}{P_g}$$

[0082]

수학식 2는 목표 성능 달성을 위한 성능 조절 값을 CPU 대역폭 파라미터로 변환하기 위한 것이다. 여기서 Q_i 는 특정한 시점(i)에서의 자원(예를 들어, 중앙 처리 장치)의 가용 시간을 의미하고, Q_{i+1} 은 다음 시점(i+1)에서의 자원의 가용 시간을 의미하며, F는 주기(us)를 의미한다. 예를 들어, 자원의 가용 시간(Q_i)이 10,000 μ s이고, 주기(F)가 100,000 μ s인 경우, 선택된 소정의 입출력 스레드(91-1)에 할당되는 자원의 대역폭은 10%(=10,000/100,000 * 100)로 결정된다.

[0084] 목표 사용량 결정부(151)는, 수학식 1 및 수학식 2에 기재된 바와 같이, 특정한 시점(i)에서의 목표 성능(P_g) 및 실제 성능(P_r)의 차이를 목표 성능으로 나누어 연산하여 성능 조절 값을 획득하고, 성능 조절 값을 특정한 시점(i)의 실제 사용량(C_i)에 부가함으로써 선택된 대상(즉, 제1 입출력 스레드(91-1))에 대한 목표 사용량(C_{i+1})을 결정할 수 있다. 이 경우, 목표 사용량 결정부(151)는 목표 성능 및 실제 성능의 차이를 목표 성능으로 나눈 값에 조절 상수(k)를 더 부가하여 성능 조절 값을 획득하고, 획득한 성능 조절 값을 실제 사용량(C_i)에 부가하여 가중한 후 다음 시점(i+1)에서의 목표 사용량(C_{i+1})을 연산할 수도 있다.

[0085] 상술한 바와 같이 목표 사용량(C_{i+1})을 결정하는 경우, 만약 수집한 실제 성능(P_r)이 목표 성능(P_g)보다 낮다면(즉, $P_r < P_g$), 성능 조절 값은 양(+)의 값을 가지게 되고, 선택된 제1 입출력 스레드(91-1)에 대한 목표 사용량(C_{i+1})은 실제 사용량(C_i)보다 높게 결정된다. 다시 말해서, 제1 입출력 스레드(91-1)의 자원 사용량을 증가시킬 것이 결정된다. 반대로 만약 수집한 실제 성능(P_r)이 목표 성능(P_g)보다 높다면(즉, $P_r > P_g$), 상술한 성능 조절 값은 음(-)의 값을 가지게 되고, 이에 따라 제1 입출력 스레드(91-1)에 대한 목표 사용량(C_{i+1})은 실제 사용량(C_i)보다 낮게 결정된다. 다시 말해서, 제1 입출력 스레드(91-1)의 자원 사용량을 감소시킬 것이 결정된다.

[0086] 실시예에 따라서, 목표 사용량 결정부(151)는 특정한 시점(i)에서의 목표 성능(P_g) 및 실제 성능(P_r)의 차이의 절대값을 연산하고 절대값을 목표 성능으로 나누어 성능 조절 값을 획득할 수도 있다. 이 경우, 목표 사용량 결정부(151)는 먼저 실제 성능(P_r) 및 목표 성능(P_g) 사이의 크기를 비교하고, 만약 실제 성능(P_r)이 목표 성능(P_g)보다 낮다면(즉, $P_r < P_g$), 성능 조절 값을 실제 사용량(C_i)에 부가하여 목표 사용량(C_{i+1})을 연산하고, 반대로 실제 성능(P_r)이 목표 성능(P_g)보다 높다면(즉 $P_r > P_g$), 성능 조절 값을 실제 사용량(C_i)으로부터 차감하여 목표 사용량(C_{i+1})을 연산할 수도 있다.

[0087] 목표 사용량 결정부(151)의 결정 결과는 제어 신호 생성부(153)로 전달될 수 있다. 제어 신호 생성부(153)는 목표 사용량 결정부(151)의 결정 결과에 대응하여 사용량의 증가 또는 감소에 대한 제어 신호를 생성하고, 이를 기반으로 선택된 입출력 스레드, 즉 제1 입출력 스레드(91-1)의 자원의 사용량을 제어할 수 있다. 즉, 제어 신호 생성부(153)의 동작에 따라서 제1 입출력 스레드(91-1)에 할당되는 자원이 변동되게 된다. 제어 신호는, 예를 들어, 프로세서(51)나 가상 머신(90-1 내지 90-N) 등에 전달되고, 이에 따라 입출력 스레드(91-1)의 자원 사용량이 증가 또는 감소된다. 일 실시예에 의하면, 제어 신호 생성부(153)는 자원 관리부(60)를 이용하여 제1 입출력 스레드(91-1)의 자원 사용량을 제어할 수도 있다. 이 경우, 제어 신호는 자원 관리부(60)로 전달되고, 자원 관리부(60)는 이에 응하여 제1 입출력 스레드(91-1)에 할당되는 자원을 제어할 수 있다.

[0088] 자원 관리부(60)는 자원 사용량이나 성능 정보의 결정을 위해 필요한 정보를 성능 감시부(130)로 제공하거나 및/또는 성능 제어부(150)의 제어에 따라서 가상 머신(90-1 내지 90-N) 또는 이에 대응하는 입출력 스레드(91-1 내지 91-N)에 할당되는 자원을 제어할 수 있다. 자원 관리부(60)는 프로세스나 스레드들의 자원의 사용을 제한

또는 격리할 수 있도록 마련된 것일 수 있으며, 커널 영역(42) 내에서 동작하는 커널 모듈로 구현될 수도 있다. 일 실시예에 의하면, 자원 관리부(60)는 리눅스 운영 체제 등에서 제공되는 컨트롤 그룹(cGroup, Control Group)을 이용하여 구현될 수도 있으나, 이에 한정되는 것은 아니다.

[0089] 상술한 성능 관리부(100)의 동작은 복수의 가상 머신(90-1 내지 90-N) 또는 복수의 가상 머신(90-1 내지 90-N) 각각에 대응하는 복수의 입출력 스레드(91-1 내지 91-N)에 대해 수행될 수 있으며, 복수의 가상 머신(90-1 내지 90-N)의 전부 또는 복수의 입출력 스레드(91-1 내지 91-N)의 전부에 대해 순차적으로 또는 임의적으로 수행될 수도 있고, 복수의 가상 머신(90-1 내지 90-N)의 일부 또는 복수의 입출력 스레드(91-1 내지 91-N)의 일부에 대해 순차적으로 또는 임의적으로 수행될 수도 있다. 또한, 상술한 성능 관리부(100)의 동작은 주기적으로 또는 비주기적으로 반복적으로 수행될 수도 있다. 예를 들어, 제1 가상 머신(90-1) 및/또는 제1 입출력 스레드(91-1)는 주기적으로 또는 비주기적으로 그 성능이 감시될 수 있으며 필요에 따라 그(들)의 자원 사용량이 변동될 수도 있다.

[0090] 상술한 바와 같이, 가상 머신(90: 90-1 내지 90-N)이 여러 작업을 수행하는 과정에서 실제 성능이 목표 성능에 미치지 못하는 경우에는, 가상 머신(91: 90-1 내지 90-N) 각각의 입출력 스레드(91: 91-1 내지 91-N)에 제공되는 자원(예를 들어, 중앙 처리 장치의 대역폭)이 조절될 수 있으며, 이에 따라 사용자 등이 요구하는 성능의 달성이 보다 신속 및 용이하게 수행될 수 있게 된다. 이와 같이 피드백 제어를 통해 자원 할당을 조절하는 과정은, 사전 프로파일링(profileing) 없이 실시간으로 수행할 수 있어, 사전 프로파일링을 위한 오버헤드를 최소화할 수 있게 된다.

[0092] 이하 상술한 성능 관리부(100)가 동작하지 않는 경우에서의 정보 처리 장치(1)의 성능 및 성능 관리부(100)가 동작하는 경우에서의 정보 처리 장치(1)의 성능을 상호 비교하여 성능 관리부(100)의 효과에 대해 구체적으로 설명하도록 한다.

[0093] 도 4는 도 1에 도시된 성능 관리부의 적용 전 및 적용 후의 동작을 설명하기 위한 제1 도이다. 도 4는 성능 관리부(100)가 동작하지 않는 환경과 동작하는 환경 하에서 제1 가상 머신(VM1) 및 제2 가상 머신(VM2)이 네트워크 작업(1024B TCP 패킷의 연속적인 송신 작업)을 수행하도록 하고 수행에 따른 스루풋(throughput)의 측정 결과를 도시한 것이다. 도 4의 y축은 스루풋(단위: pps)을 의미하고, x축은 좌측으로부터 순차적으로 성능 관리부(100)가 동작하지 않는 경우 및 동작하는 경우를 나타낸다.

[0094] 도 4에 도시된 바를 참조하면, 성능 관리부(100)가 동작하지 않는 경우에는 제1 가상 머신(VM1) 및 제2 가상 머신(VM2) 양자 모두 동일하게 대략 2000 pps의 성능으로 동작하였다. 이와 반대로 성능 관리부(100)를 동작시키고 아울러 제1 가상 머신(VM1)의 목표 성능은 10,000pps으로 설정하고 제2 가상 머신(VM2)의 목표 성능은 25,000pps로 설정한 경우, 제1 가상 머신(VM1) 및 제2 가상 머신(VM2)은 설정된 목표 성능에 따라서 차등화된 성능으로 동작하게 된다. 구체적으로, 제1 가상 머신(VM1)은 실제 성능에 비해 낮은 목표 성능에 대응하여 더 낮아진 성능(10,000pps)으로 동작하고, 제2 가상 머신(VM2)은 실제 성능에 비해 높은 목표 성능에 대응하여 더 높아진 성능(25,000pps)으로 동작하게 된다.

[0095] 도 5는 도 1에 도시된 성능 관리부의 적용 전 및 적용 후의 동작을 설명하기 위한 제2 도이다. 도 5는 성능 관리부(100)가 동작하지 않는 환경과 동작하는 환경 하에서 제1 가상 머신(VM1) 및 제2 가상 머신(VM2)이 저장 장치에 랜덤으로 16kB 파일의 쓰기 및 읽기 동작을 수행하도록 하고(디스크 작업), 대역폭을 측정한 결과를 도시한 것이다. 도 4의 y축은 대역폭(단위: MBps)을 의미하고, x축은 좌측부터 순차적으로 성능 관리부(100)가 동작하지 않는 경우 및 동작하는 경우를 나타낸다.

[0096] 도 5에 도시된 바를 참조하면, 성능 관리부(100)가 동작하지 않는 경우에는 제1 가상 머신(VM1) 및 제2 가상 머신(VM2) 양자 모두 동일한 성능(대략 220 MBps)으로 동작하였으나, 성능 관리부(100)를 동작시키고 아울러 제1 가상 머신(VM1)의 목표 성능을 100MBps으로 설정하고 제2 가상 머신(VM2)의 목표 성능은 300MBps로 설정하면 제1 가상 머신(VM1) 및 제2 가상 머신(VM2) 각각은 설정된 목표 성능에 따라서 상이하게 동작하게 됨을 알 수 있다. 구체적으로 제1 가상 머신(VM1)은 실제 성능에 비해 상대적으로 낮은 목표 성능에 대응하는 성능(100MBps)으로 동작하고, 제2 가상 머신(VM2)은 실제 성능에 비해 상대적으로 높은 목표 성능에 대응하는 성능(300MBps)으로 동작하게 됨을 알 수 있다.

[0097] 도 4 및 도 5에 도시된 바와 같이, 성능 관리부(100)의 동작 및 목표 성능의 차등적 설정에 따라서 각각의 가상 머신(VM1, VM2)은 서로 상이하게 동작하게 되며, 이에 따라 가상 머신(VM1, VM2)에 대한 자원의 적절한 분배가

가능해진다.

- [0099] 이하, 도 6을 참조하여 가상화 환경에서 성능 제어 방법의 일 실시예에 대해 설명한다.
- [0100] 도 6은 가상화 환경에서 도 1에 도시된 정보 처리 장치에 의해 수행되는 성능 제어 방법을 설명하기 위한 흐름도이다.
- [0101] 도 6에 도시된 바에 의하면, 먼저 제어 목표가 되는 입출력 스레드가 선택될 수 있다(401). 이 경우 선택되는 입출력 스레드는 복수의 입출력 스레드 중에서 어느 하나의 입출력 스레드, 일례로 i 번째 입출력 스레드(i 는 1 이상의 자연수)일 수 있다.
- [0102] 제어 목표가 되는 입출력 스레드 또는 입출력 스레드에 대응하는 가상 머신에 대한 실제 성능과, 입출력 스레드 또는 가상 머신의 자원 사용량이 획득될 수 있다(403). 또한, 실제 성능 및 사용량의 획득 이전, 이후 또는 획득과 동시에 목표 성능 역시 획득될 수 있다. 목표 성능은 사용자에 의해 입력된 것일 수도 있고, 설계자에 의해 사전에 정의된 것일 수도 있으며, 정보 처리 장치에 의해 별도로 결정된 것일 수도 있다.
- [0103] 순차적으로 목표 성능과 실제 성능이 비교될 수 있다(405). 만약 목표 성능과 실제 성능이 동일하다면(405의 YES), 후술하는 조절 값 연산 내지 목표 사용량 기반 제어 과정(407 내지 415)은 수행되지 않을 수 있다. 실시예에 따라서, 목표 성능과 실제 성능이 일정한 값(예를 들어, 오차 범위 등)의 차이만 존재하는 경우에도 성능이 실질적으로 동일하다고 판단할 수도 있다. 이 경우에는 후술하는 조절 값 연산 내지 목표 사용량 기반 제어 과정(407 내지 415)은 수행되지 않을 수 있다.
- [0104] 만약 목표 성능과 실제 성능이 상이하다면(405의 NO), 이에 응하여 성능 조절 값이 연산될 수 있다(407). 성능 조절 값은, 상술한 수학적 식 1 및 수학적 식 2에 기재된 바와 같이, 목표 성능 및 실제 성능의 차이를 목표 성능으로 나눠 획득된 값일 수도 있고, 또는 이와 같이 획득된 값에 조절 상수를 곱하여 획득된 값일 수도 있다. 이 경우에 성능 조절 값은 목표 성능 및 실제 성능에 따라서 양의 값을 가질 수도 있고 또는 음의 값을 가질 수도 있다. 만약 목표 성능이 실제 성능보다 크다면 성능 조절 값은 양의 값을 가지고, 반대로 실제 성능이 목표 성능보다 크다면 성능 조절 값은 음의 값을 가질 수 있다. 이와 같이 주어진 성능 조절 값은 실제 사용량에 부가되고 그 결과에 따라 목표 사용량이 획득될 수 있다.
- [0105] 일 실시예에 의하면, 성능 조절 값은 목표 성능 및 실제 성능의 차이의 절대 값을 목표 성능으로 나누거나 또는 목표 성능 및 실제 성능의 차이를 목표 성능으로 나눠 획득된 값의 절대값일 수도 있다. 이 경우엔 먼저 목표 성능과 실제 성능이 상호 비교된다(409).
- [0106] 만약 목표 성능이 실제 성능보다 크다면(409의 YES), 목표 사용량은 성능 조절 값을 실제 사용량에 합산하여 결정될 수 있다(411). 반대로 목표 성능이 실제 성능보다 작다면(409의 NO), 목표 사용량은 성능 조절 값을 실제 사용량으로부터 차감하여 결정될 수 있다(413).
- [0107] 어느 과정을 이용하더라도 목표 성능이 실제 성능보다 더 큰 경우엔 목표 사용량은 실제 사용량보다 더 크게 결정되고, 반대로 목표 성능이 실제 성능보다 더 작은 경우엔 목표 사용량은 실제 사용량보다 더 작게 결정된다.
- [0108] 목표 사용량이 결정되면, 목표 사용량을 기반으로 자원(예를 들어, 중앙 처리 장치)에 대한 입출력 스레드(i)의 사용량이 변경될 수 있다(415). 예를 들어, 선택된 입출력 스레드(i)에 할당되는 자원은 증가되거나 또는 감소될 수 있다. 이 경우, 별도로 커널 등에 설치된 자원 관리 인터페이스를 이용하여 입출력 스레드에 대한 자원의 할당이 제어될 수도 있다.
- [0109] 실시예에 따라서, 순차적으로 선택된 입출력 스레드(i) 외의 다른 입출력 스레드에 대한 제어 여부 또는 선택된 입출력 스레드(i)에 대한 추가 제어 여부가 결정될 수 있다(417). 만약 추가 제어 여부가 결정되면(417의 예), 소정의 입출력 스레드($i+k$, $k \geq 0$)가 선택되고 상술한 과정(401 내지 417)이 반복될 수 있다. 여기서, k 는 1일 수도 있고 또는 1 외의 다른 자연수일 수도 있다. 다시 말해서, 복수의 입출력 스레드($i+k$, $k \geq 0$) 모두가 미리 정의된 순서나 임의적인 순서에 따라 순차적으로 선택될 수도 있고, 또는 복수의 입출력 스레드($i+k$, $k \geq 0$) 중 일부의 입출력 스레드(예를 들어, 입출력 스레드($i+2$, $i=1, 3, 5$))만이 선택될 수도 있다. 실시예에 따라서, 동일한 입출력 스레드(i , $k=0$)가 다시 선택되는 것도 가능하다.
- [0110] 상술한 실시예에 따른 성능 제어 방법은 컴퓨터 장치에 의해 구동될 수 있는 프로그램의 형태로 구현될 수 있다. 여기서 프로그램은 프로그램 명령, 데이터 파일 및 데이터 구조 등을 단독으로 또는 조합하여 포함할 수

있다. 프로그램은 기계어 코드나 고급 언어 코드를 이용하여 설계 및 제작된 것일 수 있다. 프로그램은 상술한 가상 환경에서 성능 제어 방법을 구현하기 위하여 특별히 설계된 것일 수도 있고, 컴퓨터 소프트웨어 분야에서 통상의 기술자에게 기 공지되어 사용 가능한 각종 함수나 정의를 이용하여 구현된 것일 수도 있다. 또한, 컴퓨터 장치는 프로그램의 기능을 실현 가능하게 하는 프로세서나 메모리 등을 포함하여 구현된 것일 수 있으며, 필요에 따라 통신 장치를 더 포함할 수도 있다.

[0111] 상술한 성능 제어 방법을 구현하기 위한 프로그램은, 컴퓨터에 의해 판독 가능한 기록 매체에 기록될 수 있다. 컴퓨터에 의해 판독 가능한 기록 매체는, 예를 들어, 하드 디스크나 플로피 디스크와 같은 자기 디스크 저장 매체, 자기 테이프, 콤팩트 디스크나 디브이디와 같은 광 기록 매체, 플롭티컬 디스크와 같은 자기-광 기록 매체 및 롬, 램 또는 플래시 메모리 등과 같은 반도체 저장 장치 등 컴퓨터 등의 호출에 따라 실행되는 특정 프로그램을 저장 가능한 다양한 종류의 하드웨어 장치를 포함할 수 있다.

[0112] 이상 가상화 환경에서의 성능 제어 방법 및 이를 위한 정보 처리 장치의 일 실시예에 대해 설명하였으나, 가상화 환경에서의 성능 제어 방법 및 이를 위한 정보 처리 장치는 오직 상술한 실시예에만 한정되는 것은 아니다. 해당 기술 분야에서 통상의 지식을 가진 자가 상술한 실시예를 기초로 수정 및 변형하여 구현 가능한 다양한 장치나 방법 역시 상술한 가상화 환경에서 성능 제어 방법 및 이를 위한 장치의 일례가 될 수 있다. 예를 들어, 설명된 기술들이 설명된 방법과 다른 순서로 수행되거나, 및/또는 설명된 시스템, 구조, 장치, 회로 등의 구성 요소들이 설명된 방법과 다른 형태로 결합 또는 조합되거나 설명된 시스템, 구조, 장치, 회로 등의 구성 요소들이 다른 구성 요소들이 추가적으로 부가되거나 또는 설명된 시스템, 구조, 장치, 회로 등의 구성 요소들이 다른 구성 요소에 의하여 대체 또는 치환되더라도 상술한 가상 환경에서 성능 제어 방법 및 이를 위한 장치의 일 실시예가 될 수 있다.

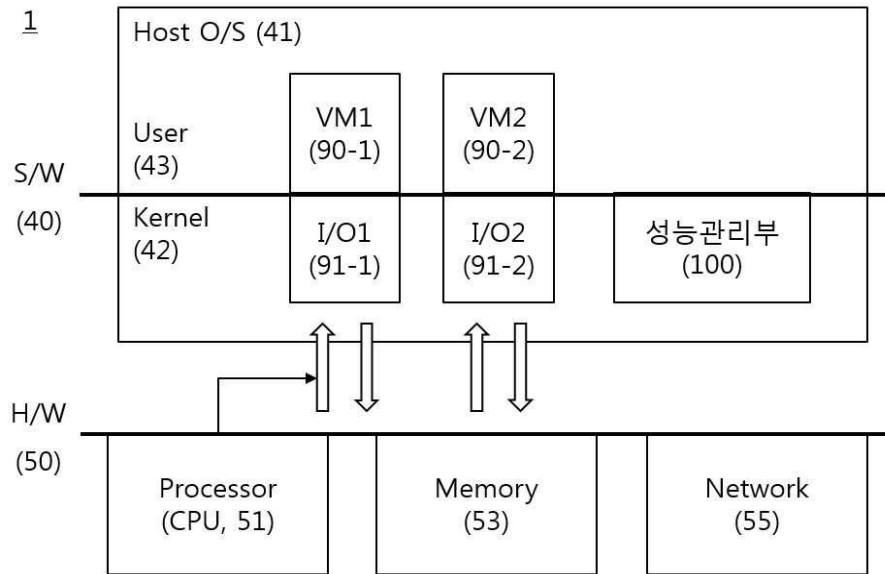
부호의 설명

[0114]

1: 정보 처리 장치 40: 소프트웨어
41: 호스트 운영 체제 50: 하드웨어
51: 프로세서 53: 메모리 장치
55: 네트워크 장치 60: 자원 관리부
90: 가상 머신 91: 입출력 스택드
100: 성능 관리부 110: 목표 성능 수신부
130: 성능 감시부 150: 성능 제어부

도면

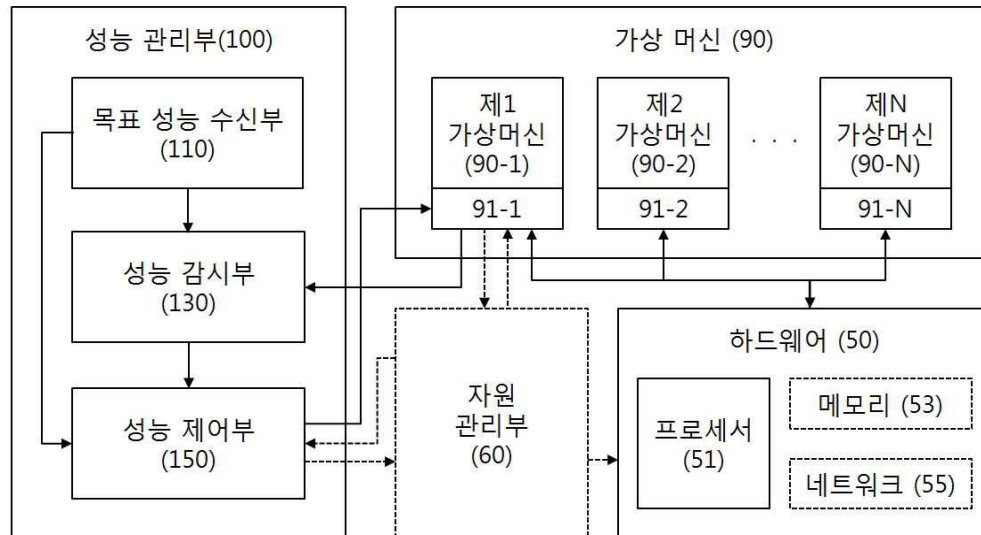
도면1



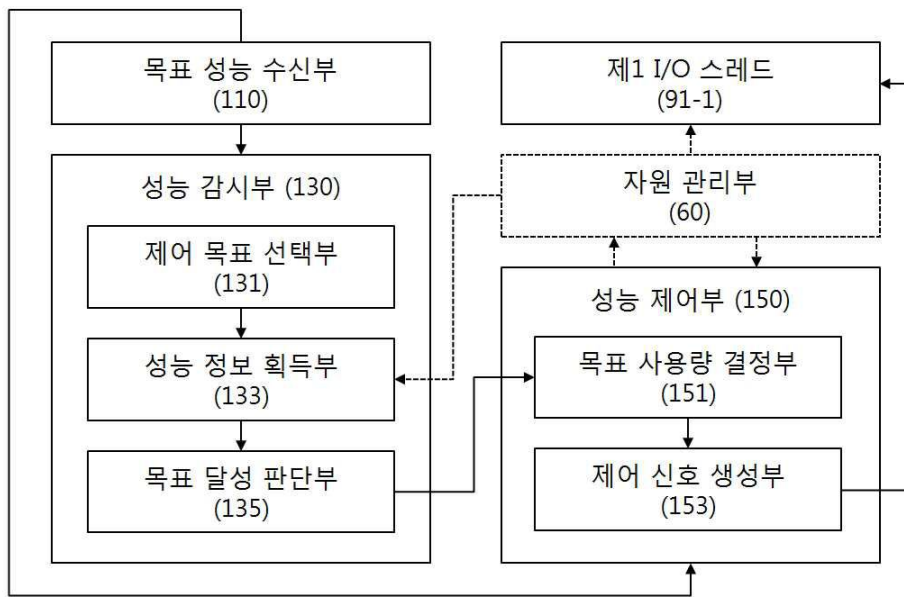
90: 90-1, 90-2

91: 91-1, 91-2

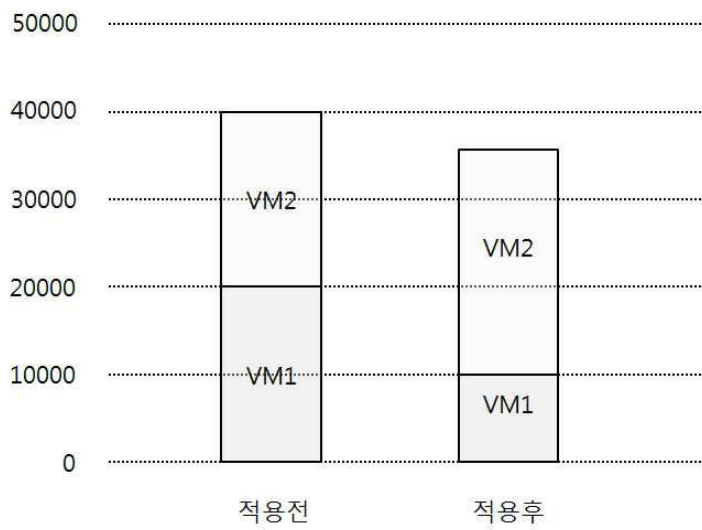
도면2



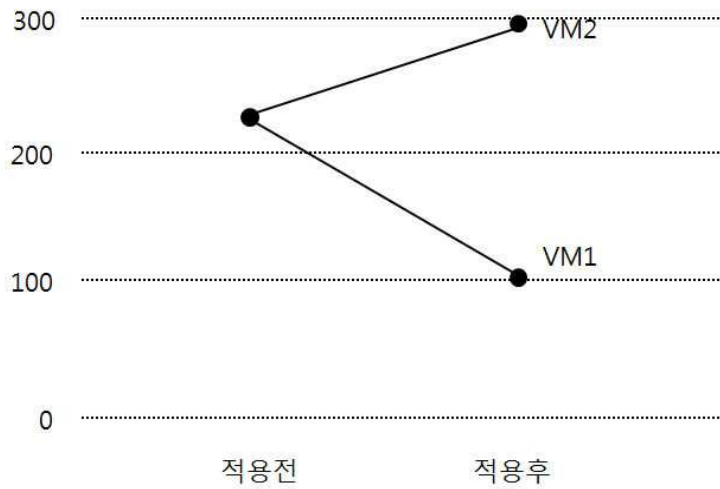
도면3



도면4



도면5



도면6

