(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2023년01월13일
(11) 등록번호 10-2488537
(24) 등록일자 2023년01월10일

(51) 국제특허분류(Int. Cl.)
G06F 9/455 (2018.01) *G06F 9/50* (2018.01)
G06T 1/20 (2018.01) *G06T 1/60* (2006.01)
(52) CPC특허분류
G06F 9/45558 (2013.01)
G06F 9/5027 (2013.01)
(21) 출원번호 10-2022-0119115
(22) 출원일자 2022년09월21일
심사청구일자 2022년09월21일
(56) 선행기술조사문헌
KR1020190070659 A*
(뒷면에 계속)

(73) 특허권자
(주)글루시스
경기도 안양시 동안구 시민대로327번길 11-31 , 5층(관양동, 파낙스알앤디센터)
세종대학교산학협력단
서울특별시 광진구 능동로 209 (군자동, 세종대학교)
(72) 발명자
노재춘
서울특별시 광진구 능동로 209, 세종대학교 대양AI센터 442호(군자동)
피아오웬유
서울특별시 광진구 광나루로22나길 12-1
(뒷면에 계속)
(74) 대리인
민영준

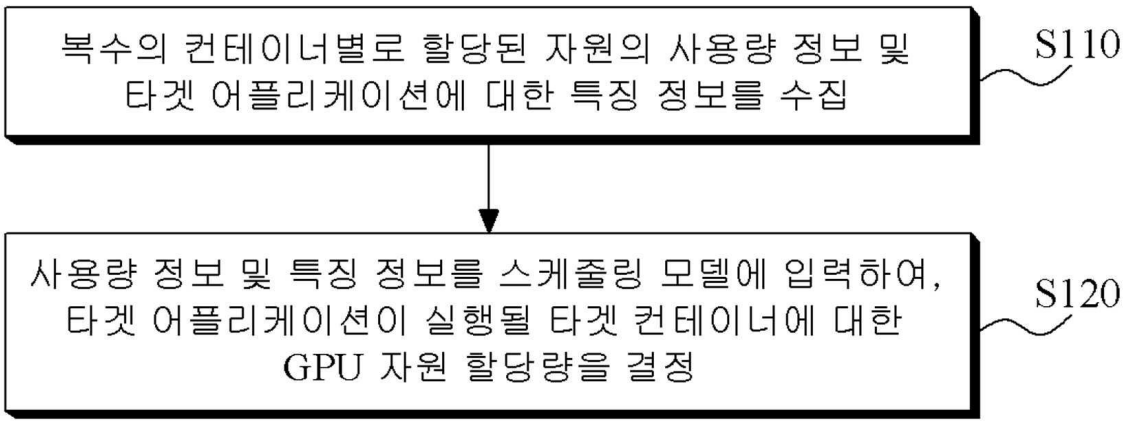
전체 청구항 수 : 총 9 항 심사관 : 최정권

(54) 발명의 명칭 **컨테이너를 이용하는 가상화 환경에서, GPU 자원을 스케줄링하는 방법**

(57) 요 약

컨테이너를 이용하는 가상화 환경에서, GPU 자원을 스케줄링하는 방법이 개시된다. 개시된 GPU 자원을 스케줄링하는 방법은 복수의 컨테이너별로 할당된 자원의 사용량 정보 및 사용자로부터 실행 요청된 타겟 어플리케이션에 대한 특징 정보를 수집하는 단계; 및 상기 사용량 정보 및 상기 특징 정보를 미리 학습된 스케줄링 모델에 입력하여, 복수의 GPU 자원 중에서, 상기 타겟 어플리케이션이 실행될 타겟 컨테이너에 대한 GPU 자원 할당량을 결정하는 단계를 포함한다.

대 표 도 - 도1



(52) CPC특허분류
G06F 9/5077 (2013.01)
G06T 1/20 (2013.01)
G06T 1/60 (2013.01)
G06F 2009/4557 (2019.08)
G06F 2009/45591 (2019.08)
G06T 2200/28 (2013.01)

(72) 발명자
김정명
경기도 광명시 양지로 7, 104동 2104호(일직동, 유-플래닛 광명역 데시앙)
박성순
경기도 군포시 수리산로 244, 995동 901호(산본동, 한양백두아파트)
김경표
광주광역시 동구 용산3길 17(용산동, 모아엘가 에듀파크)

(56) 선행기술조사문헌
KR1020190143248 A*
US20200210241 A1
KR1020220052508 A
US20220261945 A1
*는 심사관에 의하여 인용된 문헌

이 발명을 지원한 국가연구개발사업

과제고유번호	1711152425
과제번호	2021-0-00219-002
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	SW컴퓨팅산업원천기술개발
연구과제명	대용량 데이터저장 및 고속 처리 기술
기 여 율	1/1
과제수행기관명	(주)글루시스
연구기간	2022.01.01 ~ 2022.12.31

명 세 서

청구범위

청구항 1

컨테이너를 이용하는 가상화 환경에서, 컴퓨팅 장치에서 수행되는 GPU 자원을 스케줄링하는 방법에 있어서,

복수의 컨테이너별로 할당된 자원의 사용량 정보 및 사용자로부터 실행 요청된 타겟 어플리케이션에 대한 특징 정보를 수집하는 단계; 및

상기 사용량 정보 및 상기 특징 정보를 미리 학습된 스케줄링 모델에 입력하여, 복수의 GPU 자원 중에서, 상기 타겟 어플리케이션이 실행될 타겟 컨테이너에 대한 GPU 자원 할당량을 결정하는 단계; 및

상기 사용량 정보, 상기 특징 정보 및 상기 GPU 자원 할당량을 미리 학습된 데이터 분할 모델에 입력하여, 상기 타겟 어플리케이션에 대한 데이터 분할 패턴을 결정하는 단계를 포함하며,

상기 GPU 자원 할당량을 결정하는 단계는

복수의 GPU 중에서, 상기 타겟 컨테이너에 할당할 GPU를 결정하는

컨테이너를 이용하는 가상화 환경에서, GPU 자원을 스케줄링하는 방법.

청구항 2

제 1항에 있어서,

상기 사용량 정보는

CPU 사용량, GPU 사용량, CPU 메모리 사용량, GPU 메모리 사용량 중 적어도 하나를 포함하며,

상기 특징 정보는

상기 타겟 어플리케이션에 대한 데이터 크기 및 종류 중 적어도 하나를 포함하는

컨테이너를 이용하는 가상화 환경에서, GPU 자원을 스케줄링하는 방법.

청구항 3

삭제

청구항 4

삭제

청구항 5

제 1항에 있어서,

상기 데이터 분할 패턴은

상기 타겟 컨테이너에 할당된 GPU의 개수의 이하가 되도록, 상기 타겟 어플리케이션의 데이터가 분할되는 패턴인

컨테이너를 이용하는 가상화 환경에서, GPU 자원을 스케줄링하는 방법.

청구항 6

제 5항에 있어서,

상기 데이터 분할 패턴은

분할된 데이터 세그먼트의 개수 및 상기 데이터 세그먼트 각각의 크기

를 포함하는 컨테이너를 이용하는 가상화 환경에서, GPU 자원을 스케줄링하는 방법.

청구항 7

컨테이너를 이용하는 가상화 환경에서, 컴퓨팅 장치에서 수행되는 GPU 자원을 스케줄링하는 방법에 있어서,

복수의 컨테이너별로 할당된 자원의 사용량 정보 및 사용자로부터 실행 요청된 타겟 어플리케이션에 대한 특징 정보를 수집하는 단계;

상기 사용량 정보 및 상기 특징 정보를 이용하여, 복수의 GPU 자원 중에서, 상기 타겟 어플리케이션이 실행될 타겟 컨테이너에 대한 GPU 자원 할당량을 결정하는 단계; 및

상기 사용량 정보, 상기 특징 정보 및 상기 GPU 자원 할당량을 이용하여, 상기 타겟 어플리케이션에 대한 데이터 분할 패턴을 결정하는 단계

를 포함하는 컨테이너를 이용하는 가상화 환경에서, GPU 자원을 스케줄링하는 방법.

청구항 8

제 7항에 있어서,

상기 GPU 자원 할당량을 결정하는 단계는

복수의 GPU 중에서, 상기 타겟 컨테이너에 할당할 GPU를 결정하는

컨테이너를 이용하는 가상화 환경에서, GPU 자원을 스케줄링하는 방법.

청구항 9

제 8항에 있어서,

상기 데이터 분할 패턴은

상기 타겟 컨테이너에 할당된 GPU의 개수의 이하가 되도록, 상기 타겟 어플리케이션의 데이터가 분할되는 패턴인

컨테이너를 이용하는 가상화 환경에서, GPU 자원을 스케줄링하는 방법.

청구항 10

제 9항에 있어서,

상기 데이터 분할 패턴은

분할된 데이터 세그먼트의 개수 및 상기 데이터 세그먼트 각각의 크기

를 포함하는 컨테이너를 이용하는 가상화 환경에서, GPU 자원을 스케줄링하는 방법.

청구항 11

제 9항에 있어서,

상기 데이터 분할 패턴은

상기 타겟 어플리케이션의 실행을 위한 파일들이 파일 단위로 분할된 패턴인

컨테이너를 이용하는 가상화 환경에서, GPU 자원을 스케줄링하는 방법.

발명의 설명

기 술 분 야

[0001] 본 발명은 GPU 자원을 스케줄링하는 방법에 관한 것으로서, 더욱 상세하게는 컨테이너를 이용하는 가상화 환경에서, GPU 자원을 스케줄링하는 방법에 관한 것이다.

배 경 기 술

[0003] 오늘날 대표적인 클라우드 플랫폼에서는 기존에 사용하던 가상화 방법인 가상 머신 이외에도 컨테이너(container)라는 가상화 도구가 제공된다. 사용자가 원하는 자원을 요청하면, 클라우드 플랫폼은 요청된 자원이 할당된 컨테이너를 사용자에게 제공한다.

[0004] 컨테이너 기반의 가상화 방법은, 단일 컨트롤 호스트 상에서 여러 개의 고립된 리눅스 시스템을 실행하기 위한 운영 시스템 레벨 가상화 방법으로서, 운영체제 레벨 가상화라고도 불린다. 호스트 OS상에 논리적인 구획인 컨테이너가 생성되며, 컨테이너에는 어플리케이션을 작동시키기 위해 필요한 라이브러리나 어플리케이션 코드 등이 모듈화되어 할당된다.

[0005] 한편, GPU 기술이 발전함에 따라서, GPU가 그래픽 처리 분야 뿐만 아니라 기존 중앙 처리 장치(CPU)가 담당하던 응용 프로그램의 계산에도 활용되고 있으며, 컨테이너 기반의 가상화 환경에서도 복수의 GPU 자원이 이용되고 있다.

[0006] 따라서, 컨테이너는 별로 GPU 자원을 효율적으로 할당하는 스케줄링 방법이 요구되고 있다.

[0007] 관련 선행문헌으로 특허 문헌인 대한민국 등록특허 제10-2260547호, 대한민국 공개특허 제2021-0066502호, 제2022-0006360호가 있다.

발명의 내용

해결하려는 과제

[0009] 본 발명은 복수의 GPU가 이용되는 컨테이너 기반의 가상화 환경에서, GPU 자원을 효율적으로 스케줄링하는 방법을 제공하기 위한 것이다.

과제의 해결 수단

[0011] 상기한 목적을 달성하기 위한 본 발명의 일 실시예에 따르면, 복수의 컨테이너별로 할당된 자원의 사용량 정보 및 사용자로부터 실행 요청된 타겟 어플리케이션에 대한 특징 정보를 수집하는 단계; 및 상기 사용량 정보 및 상기 특징 정보를 미리 학습된 스케줄링 모델에 입력하여, 복수의 GPU 자원 중에서, 상기 타겟 어플리케이션이 실행될 타겟 컨테이너에 대한 GPU 자원 할당량을 결정하는 단계를 포함하는 컨테이너를 이용하는 가상화 환경에서, GPU 자원을 스케줄링하는 방법이 제공된다.

[0012] 또한 상기한 목적을 달성하기 위한 본 발명의 다른 실시예에 따르면, 복수의 컨테이너별로 할당된 자원의 사용량 정보 및 사용자로부터 실행 요청된 타겟 어플리케이션에 대한 특징 정보를 수집하는 단계; 상기 사용량 정보 및 상기 특징 정보를 이용하여, 복수의 GPU 자원 중에서, 상기 타겟 어플리케이션이 실행될 타겟 컨테이너에 대한 GPU 자원 할당량을 결정하는 단계; 및 상기 사용량 정보, 상기 특징 정보 및 상기 GPU 자원 할당량을 이용하여, 상기 타겟 어플리케이션에 대한 데이터 분할 패턴을 결정하는 단계를 포함하는 컨테이너를 이용하는 가상화 환경에서, GPU 자원을 스케줄링하는 방법이 제공된다.

발명의 효과

[0014] 본 발명의 일실시예에 따르면, 자원 사용량과 어플리케이션의 특징 정보에 따라 GPU 자원이 스케줄링됨으로써,

복수의 GPU 자원이 보다 효율적으로 사용될 수 있다.

[0015] 또한 본 발명의 일실시예에 따르면, 할당된 GPU 자원을 이용해 어플리케이션 데이터를 병렬로 처리함으로써, 어플리케이션 실행 속도가 향상될 수 있다.

도면의 간단한 설명

[0017] 도 1은 본 발명의 일실시예에 따른 컨테이너를 이용하는 가상화 환경에서, GPU 자원을 스케줄링하는 방법을 설명하기 위한 도면이다.

도 2 및 도 3은 본 발명의 다른 실시예에 따른 컨테이너를 이용하는 가상화 환경에서, GPU 자원을 스케줄링하는 방법을 설명하기 위한 도면이다.

도 4는 본 발명의 일실시예에 따른 high level API script 구조를 설명하기 위한 도면이다.

발명을 실시하기 위한 구체적인 내용

[0018] 본 발명은 다양한 변경을 가할 수 있고 여러 가지 실시예를 가질 수 있는 바, 특정 실시예들을 도면에 예시하고 상세한 설명에 상세하게 설명하고자 한다. 그러나, 이는 본 발명을 특정한 실시 형태에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다. 각 도면을 설명하면서 유사한 참조부호를 유사한 구성요소에 대해 사용하였다.

[0020] 본 발명은 복수의 GPU가 이용되는 컨테이너 기반의 가상화 환경에서 GPU 자원을 효율적으로 컨테이너에 할당하는 스케줄링 방법을 제안한다.

[0021] 실제 가상화 환경에서는, 복수의 GPU 개수보다 활성화된 컨테이너의 개수가 많은 것이 일반적이므로, 복수의 GPU 자원을 컨테이너에 골고루 할당하는 스케줄링이 필요할 것이다. 또한 특정 컨테이너에서 다른 컨테이너들보다 매우 많은 GPU 자원을 소비할 필요가 있다면, 이러한 컨테이너로 GPU 자원을 집중시키는 스케줄링이 필요할 것이다.

[0022] 본 발명의 일실시예는 컨테이너의 자원 사용량과 컨테이너에서 실행될 어플리케이션에 대한 특징 정보를 이용해, 컨테이너 각각에 대한 GPU 자원 할당량을 결정한다.

[0023] 본 발명의 일실시예에 따른 스케줄링 방법은 프로세서 및 메모리를 포함하는 컴퓨팅 장치에서 수행될 수 있으며, 이러한 컴퓨팅 장치는 가상화 시스템의 서버일 수 있다.

[0024] 이하에서, 본 발명에 따른 실시예들을 첨부된 도면을 참조하여 상세하게 설명한다.

[0026] 도 1은 본 발명의 일실시예에 따른 컨테이너를 이용하는 가상화 환경에서, GPU 자원을 스케줄링하는 방법을 설명하기 위한 도면이다.

[0027] 도 1을 참조하면, 본 발명의 일실시예에 따른 컴퓨팅 장치는 복수의 컨테이너별로 할당된 자원의 사용량 정보 및 사용자로부터 실행 요청된 타겟 어플리케이션에 대한 특징 정보를 수집(S110)한다.

[0028] 자원의 사용량 정보는 일실시예로서 CPU 사용량, GPU 사용량, CPU 메모리 사용량, GPU 메모리 사용량 중 적어도 하나를 포함할 수 있다. 즉, 컴퓨팅 장치는 컨테이너 별로 할당된 CPU 자원, GPU 자원, CPU 메모리 자원, GPU 메모리 자원을, 컨테이너들이 얼마나 사용하고 있는지에 대한 정보를 수집할 수 있다.

[0029] 그리고 특징 정보는 일실시예로서 타겟 어플리케이션에 대한 데이터 크기 및 종류 중 적어도 하나를 포함할 수 있다. 여기서, 종류는 카테고리를 의미할 수 있으며, 예컨대, 게임, 보안, 금융 등의 카테고리를 포함할 수 있다.

[0030] 컴퓨팅 장치는 다양한 모니터링 툴을 이용해, 자원의 사용량 정보와 특징 정보를 수집할 수 있다. 일실시예로서 컴퓨팅 장치는, 프로메테우스(Prometheus) 툴을 이용해 CPU 사용량, CPU 메모리 사용량 정보를 수집할 수 있으며, DCGM(Data Center GPU Manager) 툴을 이용해 GPU 사용량, GPU 메모리 사용량 및 어플리케이션의 특징 정보를 수집할 수 있다. 그리고 수집된 정보는 메타데이터 리포지토리(metadata repository)에 저장될 수 있다.

[0031] 본 발명의 일실시예에 따른 컴퓨팅 장치는 자원의 사용량 정보 및 특징 정보를 미리 학습된 스케줄링 모델에 입력하여, 복수의 GPU 자원 중에서, 타겟 어플리케이션이 실행될 타겟 컨테이너에 대한 GPU 자원 할당량을 결정(S120)한다.

[0032] 컴퓨팅 장치는 복수의 GPU 중에서, 타겟 컨테이너에 할당할 GPU를 결정할 수 있으며, 할당된 GPU의 자원중에서 타겟 컨테이너에 할당할 비율을 결정할 수 있다. 그리고 타겟 어플리케이션은 복수의 타겟 컨테이너에서 실행될 수 있다.

[0033] 복수의 GPU 개수보다 활성화된 컨테이너가 많은 환경에서는 1개의 GPU 자원 100%가 아닌 1개의 GPU에서 제공되는 자원의 일부가 타겟 컨테이너에 할당될 수 있다. 또는 복수의 GPU 개수보다 활성화된 컨테이너가 많은 환경이더라도, 다른 컨테이너의 자원 사용량이 많지 않으며, 타겟 컨테이너의 자원 사용량이 많은 경우에는 2개 이상의 GPU가 타겟 컨테이너에 할당될 수 있다. 이 때, 타겟 컨테이너에 할당된 복수의 GPU 각각의 자원 100%가 할당되지 않고, 복수의 GPU 각각에서 제공되는 자원의 일부가 할당될 수 있다.

[0034] 스케줄링 모델은, 자원의 사용량 정보 및 특징 정보에 대한 GPU 자원 할당량이 학습된 모델로서, 인공신경망 모델일 수 있다.

[0035] 스케줄링 모델의 학습에 이용되는 훈련 데이터는, 훈련용 사용량 정보 및 특징 정보와 함께, 훈련용 사용량 정보 및 특징 정보에 라벨링되는 정답값(ground truth)을 포함할 수 있다. 여기서 정답값은, 훈련용 사용량 정보 및 특징 정보에 라벨링되는 GPU 자원 할당량이다. 다양한 훈련용 사용량 정보와 특징 정보가 수집되는 환경에서 컨테이너에 대한 GPU 자원 할당량을 조절하면서, 정답값이 획득될 수 있다.

[0036] 본 발명의 일실시예에 따르면, 자원 사용량과 어플리케이션의 특징 정보에 따라 GPU 자원이 스케줄링됨으로써, GPU 자원이 보다 효율적으로 사용될 수 있다.

[0037] 한편, 본 발명의 일실시예는 전술된 방법에 따라 자원을 효율적으로 스케줄링할 뿐만 아니라, 스케줄링된 자원에 따라 어플리케이션 데이터를 병렬 처리하기 위해, 타겟 어플리케이션에 대한 데이터 분할 패턴을 결정할 수 있다. 데이터 분할 패턴에 따라 분할된 데이터들은 타겟 컨테이너에 할당된 복수의 GPU에서 병렬처리될 수 있다. 데이터 분할 패턴을 결정하는 방법은 도 2에서 자세히 설명된다.

[0039] 도 2 및 도 3은 본 발명의 다른 실시예에 따른 컨테이너를 이용하는 가상화 환경에서, GPU 자원을 스케줄링하는 방법을 설명하기 위한 도면이다. 그리고 도 4는 본 발명의 일실시예에 따른 high level API script 구조를 설명하기 위한 도면이다.

[0040] 도 2를 참조하면 본 발명의 일실시예에 따른 컴퓨팅 장치는 복수의 컨테이너별로 할당된 자원의 사용량 정보 및 적어도 사용자로부터 실행 요청된 타겟 어플리케이션에 대한 특징 정보를 수집(S210)한다. 컴퓨팅 장치는 활성화된 컨테이너의 개수가 변경되거나, 어플리케이션의 실행 요청이 있을 때마다 사용량 정보 및 특징 정보를 갱신할 수 있다.

[0041] 그리고 컴퓨팅 장치는 수집된 사용량 정보 및 특징 정보를 이용하여, 복수의 GPU 자원 중에서 타겟 어플리케이션이 실행될 타겟 컨테이너에 대한 GPU 자원 할당량을 결정(S220)한다. 컴퓨팅 장치는 일실시예로서, 복수의 컨테이너의 자원 사용량에 반비례하고, 타겟 어플리케이션의 크기에 비례하도록 GPU 자원 할당량을 결정할 수 있다. 예컨대 복수의 컨테이너의 자원 사용량이 적고 타겟 어플리케이션의 크기가 크다면 GPU 자원 할당량은 증가하고, 복수의 컨테이너의 자원 사용량이 많고 타겟 어플리케이션의 크기가 작다면 GPU 자원 할당량은 감소할 수 있다.

[0042] 또한 컴퓨팅 장치는 타겟 어플리케이션의 종류에 따라 GPU 자원 할당량을 결정할 수 있다. 타겟 어플리케이션이 GPU 자원을 많이 소모하는 게임 카테고리 등의 어플리케이션이라면 GPU 자원 할당량이 증가할 수 있으며, GPU 자원을 적게 소모하는 금융 카테고리 등의 어플리케이션이라면 GPU 자원 할당량이 감소할 수 있다.

[0043] 또는 컴퓨팅 장치는, 전술된 스케줄링 모델을 이용해 GPU 자원 할당량을 결정할 수 있다.

[0044] 그리고 컴퓨팅 장치는 사용량 정보, 특징 정보 및 GPU 자원 할당량을 이용하여, 타겟 어플리케이션에 대한 데이터 분할 패턴을 결정(S230)한다. 여기서, 데이터 분할 패턴은, 분할된 데이터 세그먼트의 개수 및 데이터 세그먼트 각각의 크기에 대한 패턴으로서, 타겟 컨테이너에 할당된 GPU의 개수 이하가 되도록, 타겟 어플리케이션의 데이터가 분할되는 패턴일 수 있다.

[0045] 예컨대, 타겟 컨테이너에 할당된 GPU의 개수가 3개라면, 타겟 어플리케이션의 데이터가 3개 또는 2개의 세그먼트(segment)으로 분할되는 데이터 분할 패턴이 결정될 수 있다. 또는 전술된 바와 같이, 타겟 어플리케이션이 복수의 타겟 컨테이너에서 실행되며, 복수의 타겟 컨테이너별로 1개의 GPU가 할당되었다면, 타겟 어플리케이션의 데이터가 타겟 컨테이너의 개수로 분할되는 데이터 분할 패턴이 결정될 수 있다.

[0046] 또한 데이터 분할 패턴은, 타겟 어플리케이션의 실행을 위한 파일들이 파일 단위로 분할된 패턴일 수 있다. 예컨대, 타겟 어플리케이션의 실행에 필요한 파일이 5개이며, 타겟 컨테이너에 할당된 GPU의 개수가 3개라면, 제1 및 제2파일을 포함하는 데이터 세그먼트, 제3 내지 제5파일을 포함하는 데이터 세그먼트로 분할되는 데이터 분할 패턴이 결정될 수 있다.

[0047] 한편, 단계 S230에서 컴퓨팅 장치는 일실시예로서, 사용량 정보, 특징 정보 및 GPU 자원 할당량을 미리 학습된 데이터 분할 모델에 입력하여, 타겟 어플리케이션에 대한 데이터 분할 패턴을 결정할 수 있다. 데이터 분할 모델은 자원의 사용량 정보, 특징 정보 및 GPU 자원 할당량에 대한 데이터 분할 패턴이 이 학습된 모델로서, 인공신경망 모델일 수 있다.

[0048] 도 3에 도시된 바와 같이, 사용량 정보 및 특징 정보는 스케줄링 모델(310) 및 데이터 분할 모델(320)로 입력되며, 스케줄링 모델(310)로부터 출력된 GPU 자원 할당량은 데이터 분할 모델(320)로 입력된다.

[0049] 데이터 분할 모델(320)의 학습에 이용되는 훈련 데이터는 훈련용 사용량 정보, 훈련용 특징 정보 및 훈련용 GPU 자원 할당량 함께, 이러한 정보에 라벨링되는 정답값(ground truth)을 포함할 수 있다. 여기서 정답값은, 전술된 정보에 라벨링되는 데이터 분할 패턴이다.

[0050] 타겟 어플리케이션에 대한 데이터는 데이터 분할 패턴에 따라 분할되어, 타겟 컨테이너에 할당된 GPU에서 병렬처리될 수 있다. High Level API script가 데이터 분할 패턴에 따라 타겟 어플리케이션에 대한 데이터를 복수의 데이터 세그먼트로 분할하고 MPI(Message Passing Interface) 명령어를 이용하여 데이터 세그먼트를 GPU에 병렬로 패치할 수 있다. 도 4는 High Level API script가 사용량 정보, 특징 정보 및 GPU 자원 할당량(410)에 따라 결정된 데이터 분할 패턴(420)에 의해 타겟 어플리케이션의 데이터를 4개의 데이터 세그먼트(430)로 분할하여 데이터를 처리하는 실시예가 도시되어 있다.

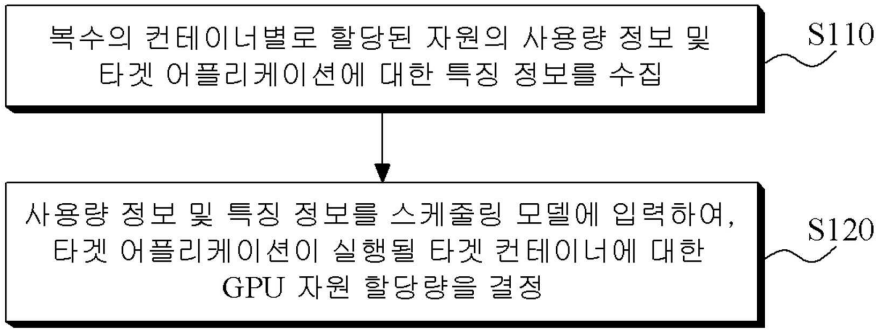
[0051] 본 발명의 일실시예에 따르면, 할당된 GPU 자원을 이용해 어플리케이션 데이터를 병렬로 처리함으로써, 어플리케이션 실행 속도가 향상될 수 있다.

[0053] 앞서 설명한 기술적 내용들은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 매체에 기록되는 프로그램 명령은 실시예들을 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(magnetic media), CD-ROM, DVD와 같은 광기록 매체(optical media), 플롭티컬 디스크(floptical disk)와 같은 자기-광 매체(magneto-optical media), 및 롬(ROM), 램(RAM), 플래시 메모리 등과 같은 프로그램 명령을 저장하고 수행하도록 특별히 구성된 하드웨어 장치가 포함된다. 프로그램 명령의 예에는 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드를 포함한다. 하드웨어 장치는 실시예들의 동작을 수행하기 위해 하나 이상의 소프트웨어 모듈로서 작동하도록 구성될 수 있으며, 그 역도 마찬가지이다.

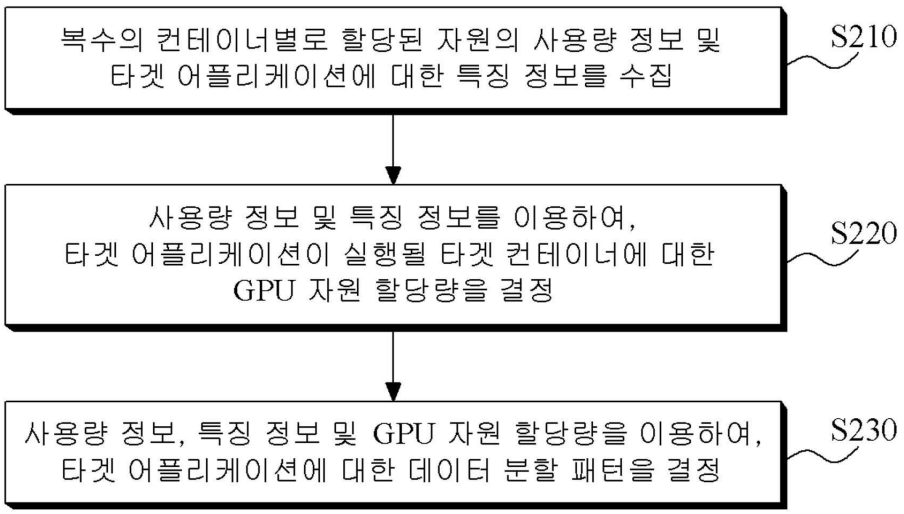
[0055] 이상과 같이 본 발명에서는 구체적인 구성 요소 등과 같은 특정 사항들과 한정된 실시예 및 도면에 의해 설명되었으나 이는 본 발명의 보다 전반적인 이해를 돕기 위해서 제공된 것일 뿐, 본 발명은 상기의 실시예에 한정되는 것은 아니며, 본 발명이 속하는 분야에서 통상적인 지식을 가진 자라면 이러한 기재로부터 다양한 수정 및 변형이 가능하다. 따라서, 본 발명의 사상은 설명된 실시예에 국한되어 정해져서는 아니되며, 후술하는 특허청구범위뿐 아니라 이 특허청구범위와 균등하거나 등가적 변형이 있는 모든 것들은 본 발명 사상의 범주에 속한다고 할 것이다.

도면

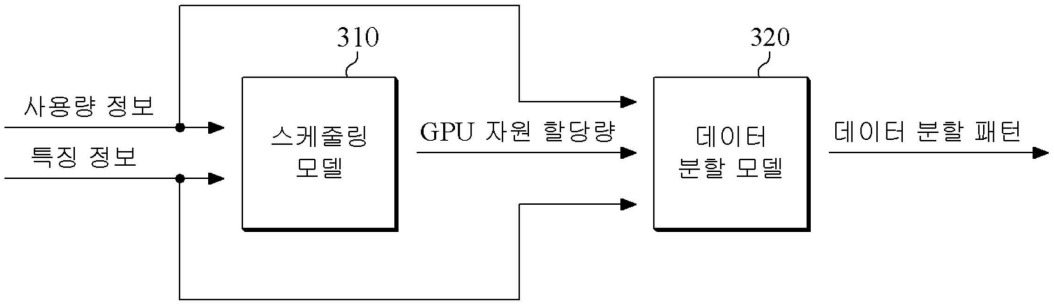
도면1



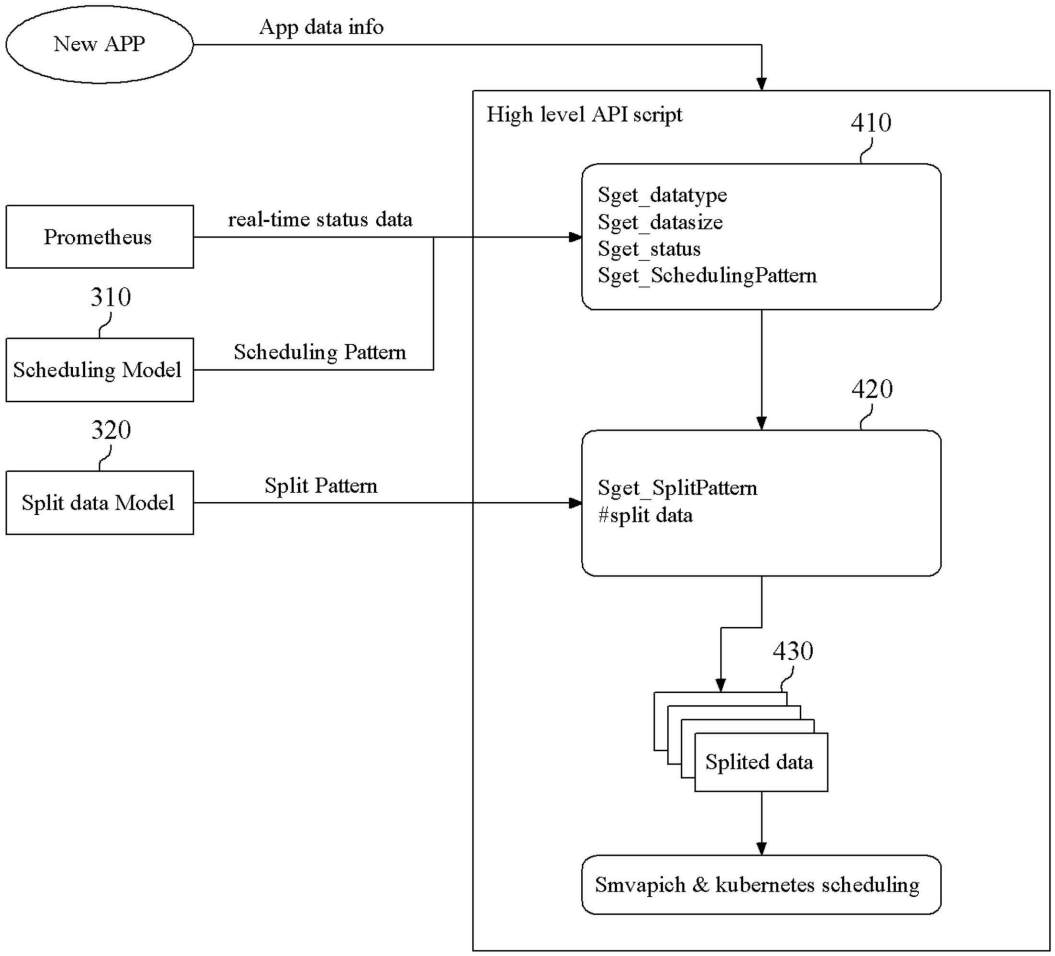
도면2



도면3



도면4


New APP
App data info
High level API script
410
Sget_datatype
Sget_datasize
Sget_status
Sget_SchedulingPattern
Prometheus
real-time status data
310
Scheduling Model
Scheduling Pattern
420
320
Split data Model
Split Pattern
Sget_SplitPattern
#split data
430
Splited data
Smvapich & kubernetes scheduling