



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2020년11월11일

(11) 등록번호 10-2177440

(24) 등록일자 2020년11월05일

(51) 국제특허분류(Int. Cl.)
G06F 9/50 (2018.01) G06F 17/10 (2006.01)
G06F 9/455 (2018.01)

(52) CPC특허분류
G06F 9/5061 (2013.01)
G06F 17/10 (2013.01)

(21) 출원번호 10-2019-0008484

(22) 출원일자 2019년01월23일

심사청구일자 2019년01월23일

(65) 공개번호 10-2020-0094838

(43) 공개일자 2020년08월10일

(56) 선행기술조사문헌

KR101772108 B1*

(뒷면에 계속)

전체 청구항 수 : 총 12 항

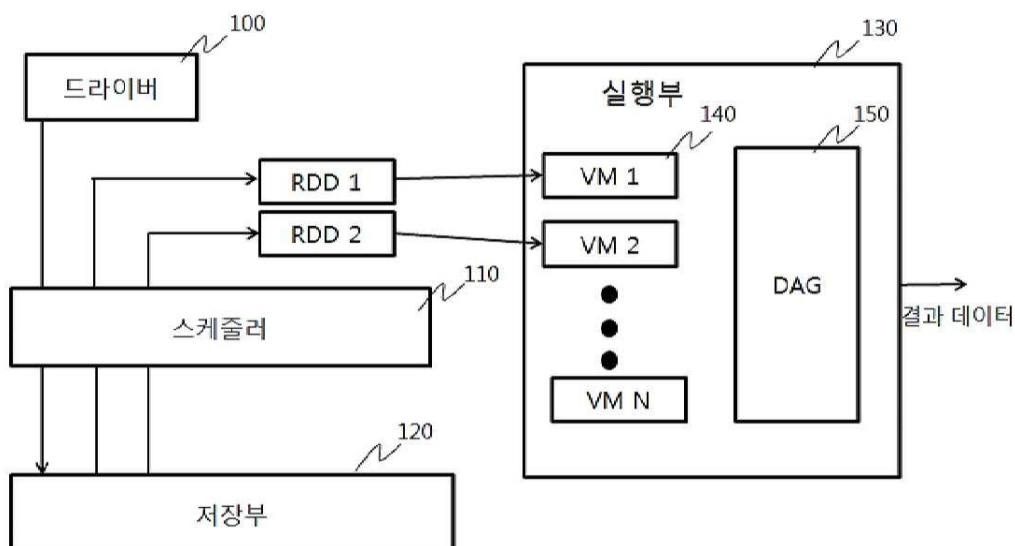
심사관 : 유진태

(54) 발명의 명칭 빅데이터 처리 장치 및 방법

(57) 요약

빅데이터 처리 장치 및 방법이 개시된다. 개시된 장치는, 빅데이터 처리에 필요한 데이터를 저장하는 저장부; 빅데이터 처리 요청에 응답하여 상기 저장부로부터 요청된 처리에 필요한 데이터를 상기 저장부로부터 로드하는 드라이버; 다수의 가상 머신을 포함하며 상기 다수의 가상 머신을 이용하여 처리 작업을 실행하는 실행부; 및 상기 실행부에서 처리 작업에 소요되는 자원을 할당하는 스케줄러를 포함하되, 상기 스케줄러는 작업의 실패 확률을 획득하고 획득된 실패 확률을 반영하여 자원-작업시간 관계 정보를 생성하며 상기 자원-작업시간 관계 정보에 기초하여 목적 작업시간을 만족시키기 위한 자원을 할당한다. 개시된 장치 및 방법에 의하면, 작업 실패를 적절히 고려하여 자원의 낭비 및 작업 지연을 발생할 수 있어서 빅데이터 처리 시스템에서의 최적화된 자원 할당이 가능한 장점이 있다.

대표도 - 도1



(52) CPC특허분류

G06F 9/45533 (2013.01)

(56) 선행기술조사문헌

JP05242103 A*

이진배 외 3명. ‘빅데이터 Hadoop Distributed File System의 자원 할당 최적화를 위한 Performance Modeling 기법’. 2018년도 한국통신학회 동계종합학술발표회 논문집, 2018.01.19., pp.1381-1382.*

KR1020140131089 A

KR1020120053857 A

*는 심사관에 의하여 인용된 문헌

이 발명을 지원한 국가연구개발사업

과제고유번호 MOIS-재난-2015-10

부처명 국민안전처

과제관리(전문)기관명 국립재난안전연구원

연구사업명 재난안전기술개발사업단

연구과제명 [행정안전부/주관]증강현실기반 재난대응 통합훈련 시뮬레이터 개발(4/4)

기 여 율 1/1

과제수행기관명 연세대학교 산학협력단

연구기간 2018.07.15 ~ 2019.07.14

공지예외적용 : 있음

명세서

청구범위

청구항 1

빅데이터 처리에 필요한 데이터를 저장하는 저장부;

빅데이터 처리 요청에 응답하여 상기 저장부로부터 요청된 처리에 필요한 데이터를 상기 저장부로부터 로드하는 드라이버;

다수의 가상 머신을 포함하며 상기 다수의 가상 머신을 이용하여 처리 작업을 실행하는 실행부; 및

상기 실행부에서 처리 작업에 소요되는 자원을 할당하는 스케줄러를 포함하되,

상기 스케줄러는 작업의 실패 확률을 획득하고 획득된 실패 확률을 반영하여 자원-작업시간 관계 정보를 생성하며 상기 자원-작업시간 관계 정보에 기초하여 목적 작업시간을 만족시키기 위한 자원을 할당하며,

상기 스케줄러는 다음의 수학적식과 같이 자원(n)과 작업 시간(T_{Est})의 관계 정보를 생성하는 것을 특징으로 하는 빅데이터 처리 장치.

$$T_{Est} = T_{init} + T_{prep} + (in\varepsilon_{vs}T_{vs}^{baseline}) + \left(i \frac{1}{n} \sum_{k=1}^{n_{unit}} M_a^k + \varepsilon_{commn} T_{commn}^{baseline} \frac{1}{s_{baseline}} \frac{s}{n} \right) + \left(in\varepsilon_{vs}T_{vs}^{baseline} \sum_{k=1}^i P_e^k \right) + \left(\varepsilon_{commn} T_{commn}^{baseline} \frac{1}{s_{baseline}} \frac{s}{n} i \sum_{k=1}^i P_e^k \right)$$

위 수학적식에서, T_{init} 는 초기화에 소요되는 시간을 의미하고, T_{prep} 는 준비 절차에 소요되는 시간을 의미하며, T_{vs} 는 변수 공유에 소요되는 시간을 의미하고, T_{commn} 는 연산 작업에 소요되는 시간을 의미하고, i 는 반복 횟수를 의

미하고, n 은 가상 머신의 수를 의미하며, ε_{vs} 는 변수 공유 단계에서의 매개 변수로서 미리 설정되는 상수이고,

ε_{commn} 는 연산 단계에서의 매개 변수로서 미리 설정되는 상수이며, $T_{commn}^{baseline}$ 는 측정된 연산 단계 시간의 평균

값으로서 측정에 의해 획득되는 변수이며, n_{unit} 은 작업당 RDD의 개수이고, $T_{vs}^{baseline}$ 은 변수 공유 단계 시간의

평균값으로서 측정에 의해 획득되는 변수이며, s 는 입력되는 데이터 사이즈이고, $s_{baseline}$ 은 입력되는 데이터 사이즈의 평균이며, P_e 는 실패 확률을 의미한다.

청구항 2

제1 항에 있어서,

상기 스케줄러는 상기 다수의 가상 머신 각각이 전송하는 보고 메시지를 이용하여 상기 실패 확률을 획득하는 것을 특징으로 하는 빅데이터 처리 장치.

청구항 3

제2 항에 있어서,

상기 보고 메시지는 작업의 실패 여부를 나타내는 실패 필드, 메시지를 송신하는 가상 머신의 번호를 나타내는

가상 머신 번호 필드 및 작업의 목적 시간 정보인 목적 시간 필드를 포함하는 것을 특징으로 하는 빅데이터 처리 장치.

청구항 4

제1항에 있어서,

상기 드라이버는 상기 저장부로부터 필요한 데이터를 RDD(Resilient Distributed Dataset) 형태로 변환하여 로드하는 것을 특징으로 하는 빅데이터 처리 장치.

청구항 5

삭제

청구항 6

제1항에 있어서,

상기 스케줄러는 다음의 수학적식을 이용하여 목적 작업 시간에 적합한 자원(n)의 수를 결정하는 것을 특징으로 하는 빅데이터 처리 장치.

$$a + bn + \frac{c}{n} \leq T_{object}$$

$$bn^2 + (a - T_{object})n + c \leq 0$$

$$a = T_{init} + T_{prep}$$

위 수학적식에서, T_{object} 는 목적 작업 시간이고, 이고,

$$b = i\varepsilon_{vs} T_{vs}^{baseline} + i\varepsilon_{vs} T_{vs}^{baseline} \sum_{k=1}^i P_{\varepsilon}^k$$

이며,

$$c = i \sum_{k=1}^{n_{unit}} M_a^k + \varepsilon_{commn} T_{commn}^{baseline} s / s_{baseline} + i\varepsilon_{commn} T_{commn}^{baseline} s \sum_{k=1}^i P_{\varepsilon}^k / s_{baseline}$$

이다.

청구항 7

제6항에 있어서,

상기 자원의 수는 상기 실행부에서 사용할 가상 머신의 수를 포함하는 것을 특징으로 하는 빅데이터 처리 장치.

청구항 8

빅데이터 처리 장치에서 수행되는 빅데이터 처리 방법으로서,

빅데이터 처리 요청에 응답하여 요청된 처리에 필요한 데이터를 로드하는 단계(a);

빅데이터 처리에 필요한 자원을 할당하는 스케줄링을 수행하는 단계(b); 및

다수의 가상 머신을 이용하여 상기 스케줄링된 자원에 기초하여 빅데이터 처리를 실행하는 단계(c)를 포함하되,

상기 단계(b)는, 작업의 실패 확률을 획득하고 획득된 실패 확률을 반영하여 자원-작업시간 관계 정보를 생성하며 상기 자원-작업시간 관계 정보에 기초하여 목적 작업시간을 만족시키기 위한 자원을 할당하며,

상기 단계(b)는 다음의 수학적식과 같이 자원(n)과 작업 시간(T_{Est})의 관계 정보를 생성하는 것을 특징으로 하는 빅데이터 처리 방법.

$$T_{Est} = T_{init} + T_{prep} + (in\varepsilon_{vs}T_{vs}^{baseline}) + \left(i \frac{1}{n} \sum_{k=1}^{n_{unit}} M_a^k + \varepsilon_{commn} T_{commn}^{baseline} \frac{1}{s_{baseline}} \frac{s}{n} \right) + \left(in\varepsilon_{vs}T_{vs}^{baseline} \sum_{k=1}^i P_e^k \right) + \left(\varepsilon_{commn} T_{commn}^{baseline} \frac{1}{s_{baseline}} \frac{s}{n} \sum_{k=1}^i P_e^k \right)$$

위 수식에서, T_{init} 는 초기화에 소요되는 시간을 의미하고, T_{prep} 는 준비 절차에 소요되는 시간을 의미하며, T_{vs} 는 변수 공유에 소요되는 시간을 의미하고, T_{comp} 는 연산 작업에 소요되는 시간을 의미하고, i 는 반복 횟수를 의

미하고, n 은 가상 머신의 수를 의미하며, ε_{vs} 는 변수 공유 단계에서의 매개 변수로서 미리 설정되는 상수이고,

ε_{commn} 는 연산 단계에서의 매개 변수로서 미리 설정되는 상수이며, $T_{commn}^{baseline}$ 는 측정된 연산 단계 시간의 평균

값으로서 측정에 의해 획득되는 변수이며, n_{unit} 은 작업당 RDD의 개수이고, $T_{vs}^{baseline}$ 은 변수 공유 단계 시간의

평균값으로서 측정에 의해 획득되는 변수이며, s 는 입력되는 데이터 사이즈이고, $s_{baseline}$ 은 입력되는 데이터 사
이즈의 평균이며, P_e 는 실패 확률을 의미한다.

청구항 9

제8항에 있어서,

상기 단계(b)는 상기 다수의 가상 머신 각각이 전송하는 보고 메시지를 이용하여 상기 실패 확률을 획득하는 것
을 특징으로 하는 빅데이터 처리 방법.

청구항 10

제9항에 있어서,

상기 보고 메시지는 작업의 실패 여부를 나타내는 실패 필드, 메시지를 송신하는 가상 머신의 번호를 나타내는
가상 머신 번호 필드 및 작업의 목적 시간 정보인 목적 시간 필드를 포함하는 것을 특징으로 하는 빅데이터 처
리 방법.

청구항 11

제8항에 있어서,

상기 단계(a)는 빅데이터 처리에 필요한 데이터를 저장하는 저장부로부터 필요한 데이터를 RDD(Resilient
Distributed Dataset) 형태로 변환하여 로드하는 것을 특징으로 하는 빅데이터 처리 방법.

청구항 12

삭제

청구항 13

제8항에 있어서,

상기 단계(b)는 다음의 수식식을 이용하여 목적 시간에 적합한 자원(n)의 수를 결정하는 것을 특징으로 하는 빅

데이터 처리 방법.

$$a + bn + \frac{c}{n} \leq T_{object}$$

$$bn^2 + (a - T_{object})n + c \leq 0$$

$$a = T_{init} + T_{prep}$$

위 수식에서, T_{object} 는 목적 작업 시간이고, a 이고,

$$b = i\varepsilon_{vs}T_{vs}^{baseline} + i\varepsilon_{vs}T_{vs}^{baseline}\sum_{k=1}^i P_s^k$$

이며,

$$c = i\sum_{k=1}^{n_{unit}} M_a^k + \varepsilon_{commn}T_{commn}^{baseline}s/s_{baseline} + i\varepsilon_{commn}T_{commn}^{baseline}s\sum_{k=1}^i P_s^k/s_{baseline}$$

이다.

청구항 14

제13항에 있어서,

상기 자원의 수는 상기 단계(c)에서 사용할 가상 머신의 수를 포함하는 것을 특징으로 하는 빅데이터 처리 방법.

발명의 설명

기술 분야

[0001] 본 발명은 빅데이터 분석 및 처리 시스템에 관한 것으로서, 더욱 상세하게는 빅데이터 처리 최적화를 위한 실행 시간 예측 및 자원 할당 방법에 관한 것이다.

배경 기술

[0003] 빅데이터가 다양한 분야에 활용되면서 빅데이터 처리의 효율성의 연구 또한 활발해지고 있다. Apache Hadoop과 Apache Spark는 BIM, AWS, Facebook, Twitter 등 전세계적으로 널리 쓰이고 있는 빅데이터 처리를 위한 오픈 분산 처리 플랫폼이다.

[0004] 빅데이터의 크기가 커질수록 서버의 규모는 커지고 있고 그에 대한 처리 비용 부담도 커지고 있다. 그에 따라 데이터 처리의 최적화 연구가 필요하게 되었고 그에 관련하여 Job 스케줄링의 방식으로 작업 시간을 낮추는 연구가 진행 되었다. 또한 처리 대상 시간을 만족시킬 수 있는 빅데이터 시스템을 구현하려면 정확한 자원 할당이 필요하고 이를 위해서는 요구된 작업에 대한 정확한 작업 실행 시간 추정이 필요하다

[0005] 스파크 기반 빅데이터 처리 시스템은 작업 요청이 들어오면 그에 필요한 데이터를 기존에 구축되어 있는 Hadoop Distribute File System에서 RDD(Resilient Distributed Dataset)라는 파일 타입으로 로드한다. 그리고 그 RDD를 변형하면서 작업을 수행하고, 그 변화 과정을 lineage라는 파일 내에 저장한다.

[0006] 스파크 기반 빅데이터 처리 시스템에서의 처리 작업은 HDFS에서 HadoopRDD로 복사해 온 RDD를 Map Transformation을 수행하여 MapRDD로 바꾸고 그 MapRDD에 Shuffle Transformation을 수행하여 ShuffleRDD로 바꾸고, 그 ShuffleRDD에 ReduceByKey Transformation을 수행하여 ReduceRDD로 바꿔 작업을 완료한다.

[0007] 이런 과정은 DAG로 구현되어 그 작업 과정을 쉽게 확인 가능하고, lineage에 저장되어 있는 변환(Transformation) 과정을 통해서 도중 오류가 발생했을 시 RDD를 다시 복구 할 수 있다.

[0008] 기존의 스파크 기반 빅데이터 처리 장치는 작업 수행 전에 작업 시간을 예측하고 이에 기초하여 최소의 자원이 할당되도록 자원 할당을 수행하며, 자원 할당을 통해 빅데이터 처리에 사용할 가상 머신의 수를 결정한다.

[0009] 그러나, 기존의 스파크 기반 빅데이터 처리 장치는 작업 실패로 인한 지연을 고려하지 않았으며 이는 부적절한 자원 할당이 이루어지는 주요한 요인으로 작용하였다. 부적절한 자원 할당은 자원의 낭비 또는 처리 작업의 지

연으로 이어지기 때문에 보다 적절한 자원 할당이 요구되는 실정이다.

발명의 내용

해결하려는 과제

[0011] 본 발명의 목적은 작업 실패를 적절히 고려하여 자원의 낭비 및 작업 지연을 발생할 수 있는 빅데이터 처리 시스템에서의 최적화된 자원 할당 방법을 제안하는 것이다.

과제의 해결 수단

[0013] 상기 목적을 달성하기 위해 본 발명의 일 측면에 따르면, 빅데이터 처리에 필요한 데이터를 저장하는 저장부; 빅데이터 처리 요청에 응답하여 상기 저장부로부터 요청된 처리에 필요한 데이터를 상기 저장부로부터 로드하는 드라이버; 다수의 가상 머신을 포함하며 상기 다수의 가상 머신을 이용하여 처리 작업을 실행하는 실행부; 및 상기 실행부에서 처리 작업에 소요되는 자원을 할당하는 스케줄러를 포함하되, 상기 스케줄러는 작업의 실패 확률을 획득하고 획득된 실패 확률을 반영하여 자원-작업시간 관계 정보를 생성하며 상기 자원-작업시간 관계 정보에 기초하여 목적 작업시간을 만족시키기 위한 자원을 할당하는 빅데이터 처리 장치가 제공된다.

[0014] 상기 스케줄러는 상기 다수의 가상 머신 각각이 전송하는 보고 메시지를 이용하여 상기 실패 확률을 획득한다.

[0015] 상기 보고 메시지는 작업의 실패 여부를 나타내는 실패 필드, 메시지를 송신하는 가상 머신의 번호를 나타내는 가상 머신 번호 필드 및 작업의 목적 시간 정보인 목적 시간 필드를 포함한다.

[0016] 상기 드라이버는 상기 저장부로부터 필요한 데이터를 RDD(Resilient Distributed Dataset) 형태로 변환하여 로드한다.

[0017] 상기 스케줄러는 다음의 수학적식과 같이 자원(n)과 작업 시간(T_{Est})의 관계 정보를 생성한다.

$$T_{Est} = T_{init} + T_{prep} + (in\varepsilon_{vs} T_{vs}^{baseline}) + \left(i \frac{1}{n} \sum_{k=1}^{n_{unit}} M_a^k + \varepsilon_{commn} T_{commn}^{baseline} \frac{1}{S_{baseline}} \frac{s}{n} \right) + \left(in\varepsilon_{vs} T_{vs}^{baseline} \sum_{k=1}^i P_e^k \right) + \left(\varepsilon_{commn} T_{commn}^{baseline} \frac{1}{S_{baseline}} \frac{s}{n} i \sum_{k=1}^i P_e^k \right)$$

[0018]

[0019] 상기 스케줄러는 다음의 수학적식을 이용하여 목적 시간에 적합한 자원(n)의 수를 결정한다.

$$a + bn + \frac{c}{n} \leq T_{object}$$

[0020]

$$bn^2 + (a - T_{object})n + c \leq 0$$

[0021]

[0022] 상기 자원의 수는 상기 실행부에서 사용할 가상 머신의 수를 포함한다.

[0023] 본 발명의 다른 측면에 따르면, 빅데이터 처리 요청에 응답하여 요청된 처리에 필요한 데이터를 로드하는 단계(a); 빅데이터 처리에 필요한 자원을 할당하는 스케줄링을 수행하는 단계(b); 및 다수의 가상 머신을 이용하여 상기 스케줄링된 자원에 기초하여 빅데이터 처리를 실행하는 단계(c)를 포함하되, 상기 단계(b)는, 작업의 실패 확률을 획득하고 획득된 실패 확률을 반영하여 자원-작업시간 관계 정보를 생성하며 상기 자원-작업시간 관계 정보에 기초하여 목적 작업시간을 만족시키기 위한 자원을 할당하는 빅데이터 처리 방법이 제공된다.

발명의 효과

[0025] 본 발명에 의하면, 작업 실패를 적절히 고려하여 자원의 낭비 및 작업 지연을 발생할 수 있어서 빅데이터 처리 시스템에서의 최적화된 자원 할당이 가능한 장점이 있다.

도면의 간단한 설명

[0027] 도 1은 본 발명의 일 실시예에 따른 스파크 기반 빅데이터 처리 장치의 구조를 도시한 도면.

도 2는 본 발명의 일 실시예에 따른 스파크 기반 빅데이터 처리 장치에서의 처리 절차를 나타낸 도면.

도 3은 본 발명의 일 실시예에 따른 스케줄러의 구조를 도시한 블록도.

도 4는 본 발명의 일 실시예에 따른 빅데이터 처리 최적화를 위한 자원 할당 방법의 전체적인 흐름을 도시한 순서도.

도 5는 본 발명의 일 실시예에 따른 가상 머신이 스케줄러에 제공하는 보고 메시지의 구조를 도시한 도면.

발명을 실시하기 위한 구체적인 내용

- [0028] 본 발명과 본 발명의 동작상의 이점 및 본 발명의 실시예에 의하여 달성되는 목적을 충분히 이해하기 위해서는 본 발명의 바람직한 실시예를 예시하는 첨부 도면 및 첨부 도면에 기재된 내용을 참조하여야만 한다.
- [0029] 이하, 첨부한 도면을 참조하여 본 발명의 바람직한 실시예를 설명함으로써, 본 발명을 상세히 설명한다. 그러나, 본 발명은 여러 가지 상이한 형태로 구현될 수 있으며, 설명하는 실시예에 한정되는 것이 아니다. 그리고, 본 발명을 명확하게 설명하기 위하여 설명과 관계없는 부분은 생략되며, 도면의 동일한 참조부호는 동일한 부재를 나타낸다.
- [0030] 명세서 전체에서, 어떤 부분이 어떤 구성요소를 “포함” 한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성요소를 제외하는 것이 아니라, 다른 구성요소를 더 포함할 수 있는 것을 의미한다. 또한, 명세서에 기재된 “...부”, “...기”, “모듈”, “블록” 등의 용어는 적어도 하나의 기능이나 동작을 처리하는 단위를 의미하며, 이는 하드웨어나 소프트웨어 또는 하드웨어 및 소프트웨어의 결합으로 구현될 수 있다.
- [0031] 도 1은 본 발명의 일 실시예에 따른 스파크 기반 빅데이터 처리 장치의 구조를 도시한 도면이다.
- [0032] 도 1을 참조하면, 본 발명의 일 실시예에 따른 스파크 기반 빅데이터 처리 장치는 드라이버(100), 스케줄러(110), 저장부(120), 실행부(130), 다수의 가상 머신(140) 및 DAG(150)를 포함한다.
- [0033] 드라이버(100)는 클라이언트로부터 빅데이터 처리 요청을 수신할 경우 처리 작업에 필요한 데이터를 저장부(120)로부터 로드하는 기능을 한다. 저장부(120)에는 처리 작업에 필요한 데이터가 미리 저장되고, 드라이버(100)는 저장부(120)로부터 필요한 데이터를 선택적으로 로드한다.
- [0034] 드라이버(100)는 처리 작업에 필요한 데이터를 그대로 복사하여 로드하는 것이 아니라 RDD(Resilient Distributed Dataset)라는 데이터 타입으로 변환한 후 로드한다.
- [0035] 드라이버(100)에 의해 로드된 RDD는 실행부(130)로 제공되며, 실행부(130)는 다수의 가상 머신(140) 및 DAG(150)를 포함한다.
- [0036] 실행부(130)는 제공된 RDD에 대해 클라이언트로부터 요청된 작업에 형태로 변환을 수행하고 처리 작업은 다수의 가상 머신(140)에 의해 이루어진다. 가상 머신(140)에 의해 이루어지는 처리 작업은 “lineage” 파일에 저장되며 작업 중간에 에러가 발생하더라도 “lineage” 파일에 기록된 정보를 이용하여 다시 복구가 가능하다.
- [0037] 각 가상 머신(140)에 의해 처리된 결과 데이터는 DAG(150)에서 최종적으로 처리된다.
- [0038] 스케줄러(110)는 실행부(130)에서 사용하는 자원을 스케줄링한다. 구체적으로 스케줄러(110)는 실행부(130)에서 처리 작업을 위해 사용하는 가상 머신의 수를 결정한다.
- [0039] 스케줄러(110)는 처리 작업을 위해 소요되는 시간을 예측하고 예측된 소요 시간에 기초하여 사용할 가상 머신의 수를 결정한다.
- [0040] 종래의 스파크 시스템은 소요되는 시간을 예측할 때 순수하게 처리에 소요되는 시간만을 예측하였으며 작업이 실패할 경우에 소요되는 시간을 고려하지 않는 문제점이 있었다. 본 발명은 스파크 시스템의 처리 작업에서 발생하는 실패를 고려하여 소요 시간을 예측하고 이에 기초하여 사용할 가상 머신의 수를 결정하도록 한다.
- [0041] 스케줄러(110)의 구체적인 구성은 별도의 도면을 참조하여 상세히 설명한다.
- [0042] 도 2는 본 발명의 일 실시예에 따른 스파크 기반 빅데이터 처리 장치에서의 처리 절차를 나타낸 도면이다.
- [0043] 도 2를 참조하면, 본 발명의 일 실시예에 따른 스파크 기반 빅데이터 처리 장치는 초기화 절차를 진행한다(200). 초기화는 로그 기록, 함수, 자원(사용하는 가상 머신의 수)과 같은 설정 값들을 초기화하는 것을 의미한다.
- [0044] 초기화가 이루어지면, 요청된 작업을 수행하기 위한 준비 작업을 수행한다(201). 준비 작업에서는 실제 작업 수

행을 위한 함수들을 로딩하고 로그 파일을 준비하며 태스크에 대한 스케줄링을 수행한다. 여기서의 스케줄링은 작업에 대한 스케줄링으로서, 앞서 설명한 스케줄링부에서의 가상 머신의 수를 할당하는 스케줄링과는 구별된다.

- [0045] 변수 공유 단계에서는 작업 중에 사용할 변수와 데이터를 각 가상 머신에 전달하는 과정이 이루어지며, 변수 공유 단계에서 가상 머신의 수를 결정하는 스케줄링이 이루어진다(203).
- [0046] 연산 단계에서는 각 가상 머신들을 이용하여 처리 연산이 이루어진다(204).
- [0047] 전체적인 작업이 위와 같이, 초기화-준비-변수 공유-연산의 순서로 이루어지기 때문에, 스파크 기반 빅데이터 처리 시스템의 작업 시간은 다음의 수학적 식 1과 같이 정의될 수 있다.

수학적 식 1

[0048]
$$T_{Est} = T_{init} + T_{prep} + T_{vs} + T_{comp}$$

- [0050] 위 수학적 식 1에 의해 작업 시간을 예측할 경우, 스파크 기반 빅데이터 처리 시스템이 별다른 예외 없이 작업을 수행한다면 비교적 정확하게 작업 시간을 예측할 수 있을 것이다.
- [0051] 위 수학적 식 1에서, T_{init} 는 초기화에 소요되는 시간을 의미하고, T_{prep} 는 준비 절차에 소요되는 시간을 의미하며, T_{vs} 는 변수 공유에 소요되는 시간을 의미하고, T_{comp} 는 연산 작업에 소요되는 시간을 의미한다.
- [0052] 그러나, 실제 스파크 기반 빅데이터 처리 시스템은 다양한 원인으로 인해 수행 실패가 발생한다. 따라서, 수학적 식 1과 같은 방식으로 작업 시간을 예측하는 것은 실패를 고려하지 않은 것이기에 정확하다고 볼 수 없다.
- [0053] 도 3은 본 발명의 일 실시예에 따른 스케줄러의 구조를 도시한 블록도이다.
- [0054] 도 3을 참조하면, 본 발명의 일 실시예에 따른 스케줄러는 실패 확률 획득부(300), 자원-작업시간 관계 정보 생성부(302) 및 자원 할당부(304)를 포함한다.
- [0055] 실패 확률 획득부(300)는 스파크 기반 빅데이터 처리 시스템의 실패 확률을 획득한다. 실패 확률은 다양한 방식으로 획득될 수 있을 것이나 바람직하게는 히스토리 처리를 통해 실패 확률을 획득할 수 있을 것이다. 과거의 작업 이력을 기초로 하여 성공적으로 실행한 작업의 수와 실패한 작업의 수에 대한 비를 이용하여 실패 확률을 획득할 수 있을 것이다. 각 작업은 다수의 반복으로 이루어지며 실패 확률 획득부(300)는 반복별 실패 확률을 획득한다.
- [0056] 물론, 실패 확률은 히스토리 분석 방식 이외에도 다양한 방식으로 획득될 수 있을 것이며, 실패 확률 획득 방식의 변경이 본 발명의 사상과 범주에 영향을 미치지 않는다는 것은 당업자에게 있어 자명할 것이다.
- [0057] 자원-작업시간 관계 정보 생성부(302)는 자원과 작업 시간과의 관계 정보를 생성한다. 작업 시간은 자원에 대한 함수이며, 자원에 따라 작업시간이 어떻게 결정되는지에 대한 관계 정보를 획득하는 것이다. 요컨대, 자원에 대한 작업 시간 함수를 생성하는 것이다.
- [0058] 스파크 기반 빅데이터 처리 시스템은 반복작업을 수행하므로 각 작업에 대한 반복 횟수를 i 라고 할 때 작업 완료까지의 실패 확률(P_{RDD})은 다음의 수학적 식 2와 같이 정의될 수 있을 것이며, 실패 확률 획득부(300)는 작업의 반복 횟수를 고려하여 다음의 수학적 식 2와 같이 실패 확률을 획득한다.

수학적 식 2

[0059]
$$P_{RDD} = P_e + P_e^2 + P_e^3 + \dots + P_e^i = \sum_{k=1}^i P_e^k$$

- [0060] 위 수학적식2에서, P_e^k 는 k 번째 반복에서의 실패 확률을 의미한다.

[0061] 본 발명은 작업 완료까지의 실패 확률인 P_{RDD} 를 고려하여 작업 시간을 예측하도록 한다. 만약 작업 실패가 발생할 경우, 다시 작업을 수행해야 한다. 이때, 초기화 및 준비 과정이 다시 진행되지는 않으며 변수 공유와 연산 작업이 다시 수행된다.

[0062] 따라서, 실패 확률을 고려한 작업 시간은 다음의 수학적 식 3과 같이 정의될 수 있다.

수학적 식 3

[0063]

$$T_{Est} = T_{Init} + T_{prep} + T_{vs} + T_{comp} + P_{RDD}(T_{vs} + T_{commn})$$

[0064] 자원-작업시간 관계 정보 생성부(302)는 수학적 식 3과 같이 작업 시간이 설정된다는 전제에서 자원과 작업 시간과의 관계 정보를 생성한다.

$$T_{Init}, T_{prep}, T_{vs}, T_{comp}$$

[0065] 는 측정을 통해 획득할 수 있는 값이며, 자원과 작업 시간과의 관계 정보는 다음의 수학적 식 4와 같이 정의될 수 있다.

수학적 식 4

[0066]

$$T_{Est} = T_{Init} + T_{prep} + (in\varepsilon_{vs}T_{vs}^{baseline}) + \left(i \frac{1}{n} \sum_{k=1}^{n_{unit}} M_a^k + \varepsilon_{commn} T_{commn}^{baseline} \frac{1}{s_{baseline}} \frac{s}{n} \right) + \left(in\varepsilon_{vs}T_{vs}^{baseline} \sum_{k=1}^i P_e^k \right) + \left(\varepsilon_{commn} T_{commn}^{baseline} \frac{1}{s_{baseline}} \frac{s}{n} \sum_{k=1}^i P_e^k \right)$$

[0067] 위 수학적 식 4에서, i 는 반복 횟수를 의미하고, n 은 가상 머신의 수를 의미하며, ε_{vs} 는 변수 공유 단계에서의 매개 변수로서 미리 설정되는 상수이고, ε_{commn} 는 연산 단계에서의 매개 변수로서 미리 설정되는 상수이며, $T_{commn}^{baseline}$ 는 측정된 연산 단계 시간의 평균값으로서 측정에 의해 획득되는 변수이며, n_{unit} 은 작업당 RDD의 개수

$$T_{vs}^{baseline}$$

이고, ε_{vs} 은 변수 공유 단계 시간의 평균값으로서 측정에 의해 획득되는 변수이며, s 는 입력되는 데이터

$$s_{baseline}$$

사이즈이고, $s_{baseline}$ 은 입력되는 데이터 사이즈의 평균이며, P_e 는 실패 확률을 의미한다.

[0068] 실패 확률을 고려한 작업 시간은 위의 수학적 식 4와 같이 자원(n)에 대한 함수로 나타낼 수 있으며, 자원 할당부(304)는 수학적 식 4를 이용하여 사용할 자원(가상 머신의 수)을 결정한다.

[0069] 자원 할당부(304)는 자원을 변수로 하는 함수인 작업 시간이 미리 설정된 목적 작업 시간 이하가 되는 자원의 수를 결정한다.

[0070] 자원 할당부(304)에서 자원의 수를 결정하기 위한 수학적 식은 다음의 수학적 식 5와 같다.

수학적 식 5

[0071]

$$a + bn + \frac{c}{n} \leq T_{object}$$

$$bn^2 + (a - T_{object})n + c \leq 0$$

위 수학적 식 5에서, T_{object} 는 목적 작업 시간이고,

위 수학적 식에서, a , b , c 는 아래와 같이 정의된다.

$$a = T_{Init} + T_{prep}$$

$$b = i\varepsilon_{vs} T_{vs}^{baseline} + i\varepsilon_{vs} T_{vs}^{baseline} \sum_{k=1}^i p_s^k$$

$$c = i \sum_{k=1}^{n_{unit}} M_a^k + \varepsilon_{commn} T_{commn}^{baseline} s / s_{baseline} + i\varepsilon_{commn} T_{commn}^{baseline} s \sum_{k=1}^i p_s^k / s_{baseline}$$

수학적 식 5와 같은 2차 부등식의 해는 하한과 상한을 가지며, 하한(n_1) 및 상한(n_2)은 다음의 수학적 식 6과 같이 정의된다.

수학적 식 6

$$n_1 = \left\lceil \frac{(a - T_{object}) + \sqrt{(a - T_{object})^2 - 4bc}}{2bc} \right\rceil$$

$$n_2 = \left\lfloor \frac{(a - T_{object}) - \sqrt{(a - T_{object})^2 - 4bc}}{2bc} \right\rfloor$$

최종적인 자원의 수는 n_1 및 n_2 사이의 값중 양의 정수인 최소값으로 결정된다. 예를 들어, n_1 이 -1이고 n_2 가 3일 경우 자원의 수는 1로 결정된다. 한편, n_1 이 5이고 n_2 가 12일 경우 자원의 수는 5로 결정된다.

이와 같은 자원 할당 방식은 최소의 자원을 사용하면서 작업 실패로 인한 추가 딜레이를 발생시키지 아니하므로 보다 효율적인 자원 할당이 가능한 장점이 있다.

도 4는 본 발명의 일 실시예에 따른 빅데이터 처리 최적화를 위한 자원 할당 방법의 전체적인 흐름을 도시한 순서도이다.

도 4를 참조하면, 우선 각 반복별 실패 확률을 획득한다(단계 400). 앞서 설명한 바와 같이 히스토리 분석을 통해 반복별 실패 확률을 획득한다.

반복별 실패 확률 정보를 획득하면, 다수의 반복으로 이루어지는 작업의 실패 확률을 획득한다(단계 402). 작업의 실패 확률은 수학적 식 2와 같이 획득할 수 있다.

작업 실패 확률이 획득되면, 획득된 작업 실패 확률을 이용하여 작업 시간과 자원에 대한 관계 정보를 생성한다(단계 404). 자원과 작업 시간에 대한 관계 정보는 수학적 식 4와 같이 생성된다.

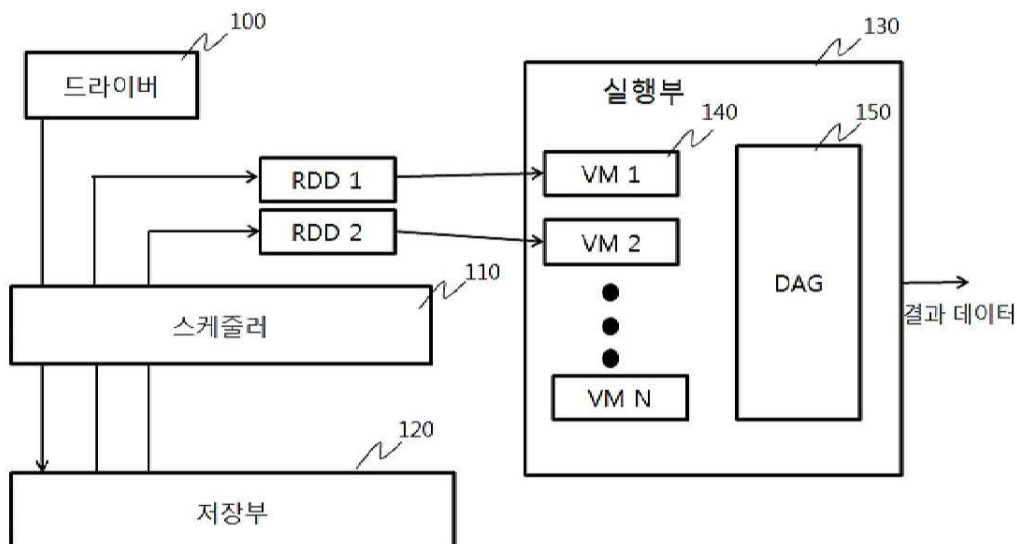
실패 확률을 고려한 작업 시간과 자원에 대한 관계 정보가 생성되면, 자원에 따른 작업 시간이 미리 설정된 작업 시간 이하가 되는 자원의 수를 결정한다(단계 406). 자원의 수는 수학적 식 5와 같은 2차 부등식에 대한 해를 구하여 결정할 수 있을 것이다.

- [0090] 도 5는 본 발명의 일 실시예에 따른 가상 머신이 스케줄러에 제공하는 보고 메시지의 구조를 도시한 도면이다.
- [0091] 도 5를 참조하면, 본 발명의 일 실시예에 따른 보고 메시지는 실패 필드(500), 가상 머신 번호 필드(502) 및 목적 시간 필드(504)를 포함한다. 처리 작업을 수행하는 가상 머신들은 스케줄러에 미리 설정된 주기 또는 미리 설정된 이벤트 발생 시마다 도 5에 도시된 바와 같은 보고 메시지를 스케줄러에 제공한다.
- [0092] 도 5와 같은 보고 메시지는 단독 메시지로 스케줄러에 제공될 수도 있으며 IP 레이어, TCP 레이어, MAC 레이어 등의 옵션 필드에 적용되어 전달될 수도 있을 것이다.
- [0093] 실패 필드(500)에는 요청된 작업이 실패 여부에 대한 정보가 기록된다. 가상 머신 번호 필드(502)에는 보고 메시지를 송신하는 가상 머신의 번호 정보가 기록된다. 목적 시간 필드(504)에는 작업을 완료해야 하는 목적 시간에 대한 정보가 기록된다.
- [0094] 스케줄러는 다수의 가상 머신들로부터 전송되는 보고 메시지를 분석하여 실패한 작업과 성공한 작업의 비를 산출하고 이를 기초로 하여 실패 확률을 획득한다. 실패 확률은 지속적으로 갱신되며, 실패 확률이 갱신될 경우 자원-작업시간 관계 정보 역시 갱신되며, 갱신된 관계 정보에 기초한 새로운 자원 할당이 이루어진다.
- [0095] 본 발명에 따른 방법은 컴퓨터에서 실행 시키기 위한 매체에 저장된 컴퓨터 프로그램으로 구현될 수 있다. 여기서 컴퓨터 판독가능 매체는 컴퓨터에 의해 액세스 될 수 있는 임의의 가용 매체일 수 있고, 또한 컴퓨터 저장 매체를 모두 포함할 수 있다. 컴퓨터 저장 매체는 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈 또는 기타 데이터와 같은 정보의 저장을 위한 임의의 방법 또는 기술로 구현된 휘발성 및 비휘발성, 분리형 및 비분리형 매체를 모두 포함하며, ROM(판독 전용 메모리), RAM(랜덤 액세스 메모리), CD(컴팩트 디스크)-ROM, DVD(디지털 비디오 디스크)-ROM, 자기 테이프, 플로피 디스크, 광데이터 저장장치 등을 포함할 수 있다.
- [0096] 본 발명은 도면에 도시된 실시예를 참고로 설명되었으나 이는 예시적인 것에 불과하며, 본 기술 분야의 통상의 지식을 가진 자라면 이로부터 다양한 변형 및 균등한 타 실시예가 가능하다는 점을 이해할 것이다.
- [0097] 따라서, 본 발명의 진정한 기술적 보호 범위는 첨부된 청구범위의 기술적 사상에 의해 정해져야 할 것이다.

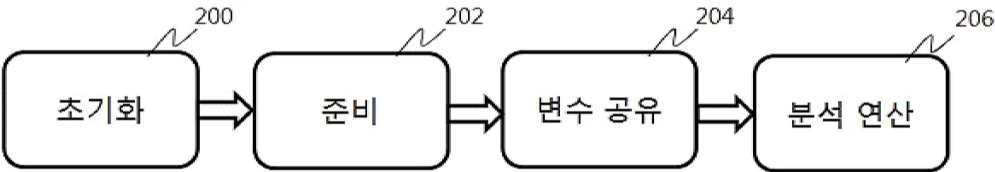
부호의 설명

도면

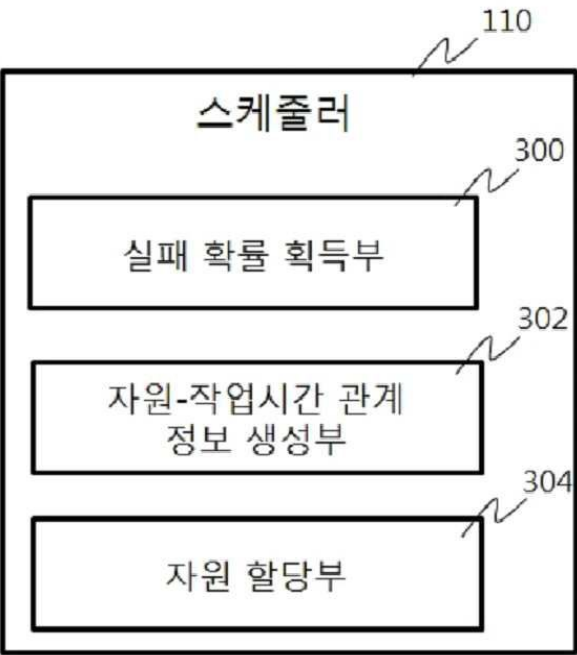
도면1



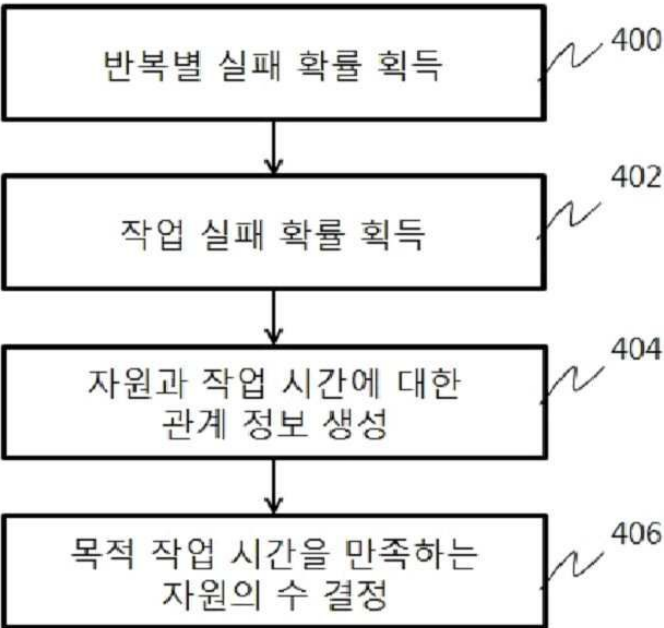
도면2



도면3



도면4



도면5

