



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2016-0076983
(43) 공개일자 2016년07월01일

- | | |
|--|--|
| <p>(51) 국제특허분류(Int. Cl.)
G06F 17/50 (2006.01) G06F 17/00 (2006.01)</p> <p>(52) CPC특허분류
G06F 17/50 (2013.01)
G06F 17/00 (2013.01)</p> <p>(21) 출원번호 10-2015-0183122</p> <p>(22) 출원일자 2015년12월21일
심사청구일자 없음</p> <p>(30) 우선권주장
14307166.0 2014년12월23일
유럽특허청(EPO)(EP)</p> | <p>(71) 출원인
다솔 시스템즈
프랑스 78140 벨리지 빌라크브레 뒤 마르셀 다솔 10</p> <p>(72) 발명자
알 마리-엘렌
프랑스 13510 에질 뒤 장 디올루페 16</p> <p>몽따나 니콜라
프랑스 13770 브넬르 앙빠스 데 그랑주 알레 졸리 페르프 8</p> <p>(74) 대리인
특허법인코리아나</p> |
|--|--|

전체 청구항 수 : 총 15 항

(54) 발명의 명칭 제어 포인트들의 그리드에 의해 정의된 3D 모델링된 오브젝트

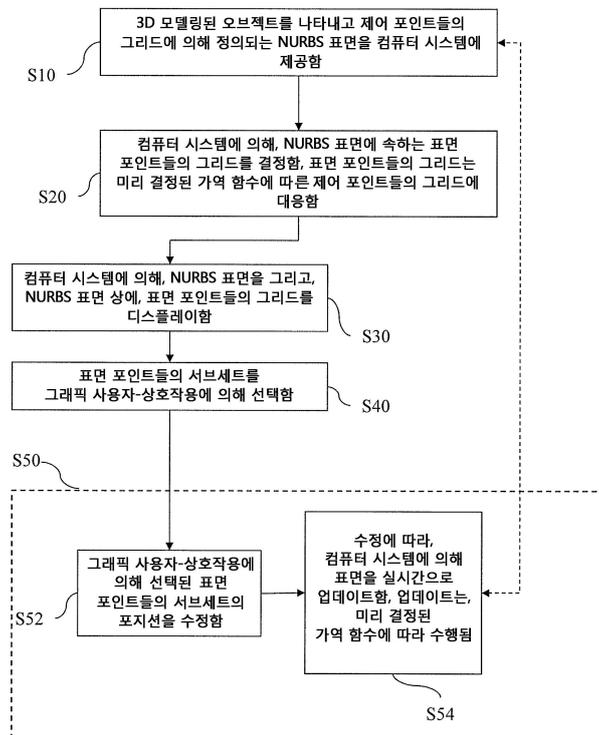
(57) 요약

3D 모델링된 오브젝트를 나타내고 제어 포인트들의 그리드에 의해 정의되는 NURBS 표면을 컴퓨터 시스템에 제공하는 단계 (S10); 컴퓨터 시스템에 의해, NURBS 표면에 속하는 표면 포인트들의 그리드를 결정함, 표면 포인트들의 그리드는 미리 결정된 가역 함수에 따른 제어 포인트들의 그리드에 대응함

로서, 표면 포인트들의 그리드는 미리 결정된 가역 함수에 따른 제어 포인트들의 그리드에 대응하는, 그 결정하

(뒷면에 계속)

대표도 - 도1



는 단계 (S20); 컴퓨터 시스템에 의해, NURBS 표면을 그리고, NURBS 표면 상에, 표면 포인트들의 그리드를 디스플레이하는 단계 (S30); 표면 포인트들의 서브셋을 그래픽 사용자-상호작용에 의해 선택하는 단계 (S40); 그래픽 사용자-상호작용에 의해 선택된 표면 포인트들의 서브셋의 위치션을 수정하는 단계 (S52), 그리고 이에 따라, 컴퓨터 시스템에 의해 표면을 실시간으로 업데이트하는 단계 (S54) 를 포함하고, 업데이트는 미리 결정된 가역 함수에 따라 수행되는, 3D 모델링된 오브젝트를 설계하기 위한 컴퓨터 구현 방법이 제공된다.

이러한 방법은 3D 모델링된 오브젝트의 설계를 개선시킨다.

(52) CPC특허분류

G06F 17/5004 (2013.01)

명세서

청구범위

청구항 1

3D 모델링된 오브젝트를 설계하기 위한 컴퓨터 구현 방법으로서,

상기 컴퓨터 구현 방법은,

- 상기 3D 모델링된 오브젝트를 나타내고 제어 포인트들의 그리드에 의해 정의되는 NURBS 표면을 컴퓨터 시스템에 제공하는 단계 (S10);
- 상기 컴퓨터 시스템에 의해, 상기 NURBS 표면에 속하는 표면 포인트들의 그리드를 결정하는 단계 (S20)로서, 상기 표면 포인트들의 그리드는 미리 결정된 가역 함수에 따른 상기 제어 포인트들의 그리드에 대응하는, 상기 결정하는 단계 (S20);
- 상기 컴퓨터 시스템에 의해, 상기 NURBS 표면을 그리고, 상기 NURBS 표면 상에, 상기 표면 포인트들의 그리드를 디스플레이하는 단계 (S30);
- 상기 표면 포인트들의 서브셋을 그래픽 사용자-상호작용에 의해 선택하는 단계 (S40);
- 상기 그래픽 사용자-상호작용에 의해 선택된 상기 표면 포인트들의 서브셋의 위치를 수정하는 단계 (S52), 그리고 이에 따라, 상기 컴퓨터 시스템에 의해 표면을 실시간으로 업데이트하는 단계 (S54)

를 포함하고,

상기 업데이트는 상기 미리 결정된 가역 함수에 따라 수행되는, 3D 모델링된 오브젝트를 설계하기 위한 컴퓨터 구현 방법.

청구항 2

제 1 항에 있어서,

상기 표면 포인트들의 그리드는 Gréville 포인트들의 그리드인, 3D 모델링된 오브젝트를 설계하기 위한 컴퓨터 구현 방법.

청구항 3

제 1 항 또는 제 2 항에 있어서,

상기 제어 포인트들의 그리드를 업데이트하는 단계 (S54)는, 선택된 상기 표면 포인트들의 서브셋의 위치가 수정된 대로 있다는 제약 하에서, 상기 NURBS 표면의 기하학적 변형에 제제를 가하는 프로그램을 최적화함으로써 수행되는, 3D 모델링된 오브젝트를 설계하기 위한 컴퓨터 구현 방법.

청구항 4

제 3 항에 있어서,

상기 프로그램은, 각각의 제어 포인트 각각에 대해, 상기 업데이트 전의 각각의 제어 포인트와 상기 업데이트 후의 각각의 제어 포인트 사이의 거리에 제제를 가함으로써 상기 NURBS 표면의 기하학적 변형에 제제를 가하는, 3D 모델링된 오브젝트를 설계하기 위한 컴퓨터 구현 방법.

청구항 5

제 4 항에 있어서,

상기 프로그램은, 각각의 제어 포인트 각각에 대해, 선택된 상기 표면 포인트들의 서브셋과 각각의 제어 포인

트에 대응하는 표면 포인트 사이의 거리의 증가 함수인 가중치를 이용하여 상기 업데이트 전의 각각의 제어 포인트와 상기 업데이트 후의 각각의 제어 포인트 사이의 거리에 제재를 가하는, 3D 모델링된 오브젝트를 설계하기 위한 컴퓨터 구현 방법.

청구항 6

제 5 항에 있어서,

상기 컴퓨터 구현 방법은, 상기 그래픽 사용자-상호작용에 의해 상기 증가 함수를 파라미터화하는 단계를 더 포함하는, 3D 모델링된 오브젝트를 설계하기 위한 컴퓨터 구현 방법.

청구항 7

제 5 항 또는 제 6 항에 있어서,

상기 증가 함수는, 선택된 상기 표면 포인트들의 서브세트에서의 표면 포인트에 대응하는 각각의 제어 포인트 각각에 대해 1 과 동일하고, 선택된 비-코너 표면 포인트 상에 센터링된 상기 표면 포인트들의 그리드에서 가장 큰 정방형에 속하는 표면 포인트, 또는 선택된 코어 표면 포인트를 포함하는 상기 표면 포인트들의 그리드의 1/4 에 속하는 표면 포인트에 대응하는 다른 각각의 제어 포인트 각각에 대해 엄격히 1 보다 더 높은 제 1 의 미리 결정된 수와 동일하며, 다른 각각의 제어 포인트 각각에 대해 상기 제 1 의 미리 결정된 수의 10배보다 더 높은 제 2 의 미리 결정된 수와 동일한 표시 함수인, 3D 모델링된 오브젝트를 설계하기 위한 컴퓨터 구현 방법.

청구항 8

제 7 항에 있어서,

상기 컴퓨터 구현 방법은 상기 그래픽 사용자-상호작용에 의해 상기 제 1 의 미리 결정된 수를 설정하는 단계를 포함하는, 3D 모델링된 오브젝트를 설계하기 위한 컴퓨터 구현 방법.

청구항 9

제 3 항 내지 제 8 항 중 어느 한 항에 있어서,

상기 프로그램은 상기 업데이트 후 그리고 상기 업데이트 전의 상기 제어 포인트들 각각에 좌우되는 돌출 에너지인, 3D 모델링된 오브젝트를 설계하기 위한 컴퓨터 구현 방법.

청구항 10

제 9 항에 있어서,

상기 프로그램은 다음과 같은 타입의 것이고:

$$\sum_{i=0}^n \sum_{j=0}^m \mu_{ij} \|P_{ij} - Q_{ij}\|^2$$

상기 P_{ij} 및 Q_{ij} 는 업데이트 후 그리고 업데이트 전 각각의 제어 포인트들인, 3D 모델링된 오브젝트를 설계하기 위한 컴퓨터 구현 방법.

청구항 11

제 9 항 또는 제 10 항에 있어서,

상기 프로그램은 라그랑주 해결법 (Lagrange solving) 으로 해결되는, 3D 모델링된 오브젝트를 설계하기 위한 컴퓨터 구현 방법.

청구항 12

제 11 항에 있어서,

상기 라그랑주 해결법은, 입력으로서 선택된 상기 표면 포인트들의 서브세트의 위치션을 취급하고 업데이트된

제어 포인트들을 출력하는 어픽스 함수 (affix function) 의 파라미터들을 결정하는 것, 그리고 그 후에 상기 어픽스 함수를 선택된 상기 표면 포인트들의 서브세트의 포지션에 수정된 대로 적용하는 것을 포함하는, 3D 모델링된 오브젝트를 설계하기 위한 컴퓨터 구현 방법.

청구항 13

제 12 항에 있어서,

선택된 상기 표면 포인트들의 서브세트의 포지션을 수정하는 단계 (S52) 및 상기 제어 포인트들의 그리드를 업데이트하는 단계 (S54) 는 동일한 선택된 상기 표면 포인트의 서브세트를 이용하여 반복되고, 상기 어픽스 함수의 파라미터들은 초기 반복시 컴퓨팅 및 저장되고 그 후에 후속 반복들 전반에 걸쳐 추출되는, 3D 모델링된 오브젝트를 설계하기 위한 컴퓨터 구현 방법.

청구항 14

제 1 항 내지 제 13 항 중 어느 한 항에 기재된 컴퓨터 구현 방법을 수행하기 위한 명령들을 포함하는, 컴퓨터 프로그램.

청구항 15

그래픽 사용자 인터페이스 및 메모리에 커플링된 프로세서를 포함하는 CAD 시스템으로서, 상기 메모리는 제 14 항에 기재된 컴퓨터 프로그램을 기록한, CAD 시스템.

발명의 설명

기술 분야

[0001] 본 발명은 특히 컴퓨터 지원 설계 (CAD) 의 분야에 관한 것이고, 더욱 구체적으로는 그래픽 사용자-상호작용에 의해 3D 모델링된 오브젝트를 설계하기 위한 방법, 프로그램 및 시스템에 관한 것이다.

배경 기술

[0002] 오브젝트들의 설계, 엔지니어링 및 제조를 위해 다수의 시스템들 및 프로그램들이 시중에 제공되어 있다. CAD 는 컴퓨터 지원 설계 (Computer-Aided Design) 의 두문자어 (acronym) 이며, 예를 들어, 오브젝트를 설계하기 위한 소프트웨어 솔루션들에 관한 것이다. CAE 는 컴퓨터 지원 엔지니어링 (Computer-Aided Engineering) 의 두문자어이며, 예를 들어, 미래의 제품의 물리적 거동을 시뮬레이션하기 위한 소프트웨어 솔루션들에 관한 것이다. CAM 은 컴퓨터 지원 제조 (Computer-Aided Manufacturing) 의 두문자어이며, 예를 들어, 제조 프로세스 및 동작들을 정의하기 위한 소프트웨어 솔루션들에 관한 것이다. 이러한 컴퓨터 지원 설계 시스템들에서, 그래픽 사용자 인터페이스는 기법의 효율과 관련하여 중요한 역할을 한다. 이들 기법들은 제품 수명주기 관리 (Product Lifecycle Management; PLM) 시스템들 내에 임베딩될 수도 있다. PLM 은 회사들이 확장 기업의 개념 전체에 걸쳐서, 제품 데이터를 공유하고, 공통 프로세스들을 적용하고, 구상에서부터 제품들의 수명의 종료까지 제품들의 개발을 위한 기업의 지식에 영향력을 행사하는 것을 돕는 경영 전략을 말한다.

[0003] 다솔 시스템즈 (DASSAULT SYSTEMES) 에 의해 (CATIA, ENOVIA 및 DELMIA 라는 상표들로) 제공되는 PLM 솔루션들은, 제품 엔지니어링 지식을 조직화하는 엔지니어링 허브, 제조 엔지니어링 지식을 관리하는 제조 허브, 그리고 엔지니어링 및 제조 허브들 양쪽 모두로의 기업 통합 및 연결을 가능하게 하는 기업 허브를 제공한다. 다 함께 이 시스템은 제품들, 프로세스들, 리소스들을 링크하는 개방형 오브젝트 모델을 전달하여 최적화된 제품 정의, 제조 준비, 생산 및 서비스를 추진하는 동적인 지식 기반 제품 창작 및 판정 지원을 가능하게 한다.

[0004] CAD 는 종종 3D 모델링된 오브젝트들을 나타내기 위한 표면들을 수반한다. 표면 모델링은 클래스 A 설계를 포함하여 특히 스타일링 설계에서 사용되고, 여기서 동기부여들은 제조된 제품의 미적 양태를 포함한다. 통상적인 표면들은 베지어, B-스플라인 그리고 더욱 일반적으로는 NURBS (non-uniform rational B-spline) 곡선들 및 표면 모델들의 수학적 방법들에 의해 제공된다. 이들 모델들 모두는 "제어 포인트들" 의 매우 대중적인 개념을 제공한다. 표면은 포인트들의 그리드에 의해 정의되고 이들 포인트들을 적절히 이동시키는 것은 표면의 형상을 변경시킨다. 통상적인 참조문헌들은 다음과 같다:

- [0005] • The NURBS book, L. Piegl, W. Tiler, Springer Science, 1997; 그리고
- [0006] • Curves and surfaces for computer aided geometric design: a practical guide, G. Farin, Hardcover, 1996.
- [0007] 기존 시스템들은 표면의 수정을 위해 사용자에게 의한 쉬운 상호작용적 조작들을 가능하게 하는 것을 목표로 한다. 정의에 의하면, 쉬운 조작은 직관적이고 예측가능하여, 초보자가 표면의 형상을 변경시키는 방법을 신속히 이해한다는 것과 숙련된 사용자가 조작으로부터 발생하는 표면을 예상할 수 있다는 것을 의미한다.
- [0008] NURBS 표면들을 형상화하기 위한 일부 방법들은 기하학적 제약들 또는 물리적 거동을 수반한다. 기하학적 제약 방법들은 새로운 제어 포인트들을 컴퓨팅하여 변형된 표면이 사용자 정의된 지오메트리 (포인트, 포인트들의 세트, 곡선) 를 충족시키도록 한다. 물리적 거동은 사용자가 표면 상에 힘을 인가하여 탄성 재료처럼 거동하는 것이다. 참조문헌들은 다음을 포함한다:
- [0009] • Dynamic NURBS with Geometric Constraints for Interactive Sculpting, D. Terzopoulos, H. Qin, ACM Transactions on Graphics, 13(2), April, 1994, 103-136; 그리고
- [0010] • Modifying the shape of NURBS surfaces with geometric constraints, S.M. Hu, Y.F. Li, T. Ju, X. Zhu, Computer Aided Design, 33 (2001) 903-912.

발명의 내용

해결하려는 과제

- [0011] 이러한 맥락 내에서, 3D 모델링된 오브젝트를 설계하기 위한 개선된 솔루션에 대한 필요성이 여전히 존재한다.

과제의 해결 수단

- [0012] 따라서, 3D 모델링된 오브젝트를 설계하기 위한 컴퓨터 구현 방법이 제공된다. 이 방법은, 3D 모델링된 오브젝트를 나타내고 제어 포인트들의 그리드에 의해 정의되는 NURBS 표면을 컴퓨터 시스템에 제공하는 단계를 포함한다. 이 방법은 또한, 컴퓨터 시스템에 의해, NURBS 표면에 속하는 표면 포인트들의 그리드를 결정하는 단계를 포함한다. 표면 포인트들의 그리드는 미리 결정된 가역 함수에 따른 제어 포인트들의 그리드에 대응한다. 이 방법은 또한, 컴퓨터 시스템에 의해, NURBS 표면을 그리고, NURBS 표면 상에, 표면 포인트들의 그리드를 디스플레이하는 단계를 포함한다. 이 방법은 또한 표면 포인트들의 서브세트를 그래픽 사용자-상호작용에 의해 선택하는 단계를 포함한다. 이 방법은 또한 그래픽 사용자-상호작용에 의해 선택된 표면 포인트들의 서브세트의 위치를 수정하는 단계, 그리고 이에 따라, 컴퓨터 시스템에 의해 표면을 실시간으로 업데이트하는 단계를 포함하고, 업데이트는 미리 결정된 가역 함수에 따라 수행된다.
- [0013] 이 방법은 다음 중 하나 이상을 포함할 수도 있다:
- [0014] - 표면 포인트들의 그리드는 Gréville 포인트들의 그리드이다;
- [0015] - 제어 포인트들의 그리드를 업데이트하는 단계는, 선택된 표면 포인트들의 서브세트의 위치가 수정된 대로 있다는 제약 하에서, NURBS 표면의 기하학적 변형에 제재를 가하는 프로그램을 최적화함으로써 수행된다;
- [0016] - 프로그램은, 각각의 제어 포인트 각각에 대해, 업데이트 전의 각각의 제어 포인트와 업데이트 후의 각각의 제어 포인트 사이의 거리에 제재를 가함으로써 NURBS 표면의 기하학적 변형에 제재를 가한다;
- [0017] - 프로그램은, 각각의 제어 포인트 각각에 대해, 선택된 표면 포인트들의 서브세트와 각각의 제어 포인트에 대응하는 표면 포인트 사이의 거리의 증가 함수인 가중치를 이용하여 업데이트 전의 각각의 제어 포인트와 업데이트 후의 각각의 제어 포인트 사이의 거리에 제재를 가한다;
- [0018] - 이 방법은, 그래픽 사용자-상호작용에 의해 증가 함수를 파라미터화하는 단계를 더 포함한다;
- [0019] - 증가 함수는, 선택된 표면 포인트들의 서브세트에서의 표면 포인트에 대응하는 각각의 제어 포인트 각각에 대해 1 과 동일하고, 선택된 비-코너 표면 포인트 상에 센터링된 표면 포인트들의 그리드에서 가장 큰 정방형에 속하는 표면 포인트, 또는 선택된 코너 표면 포인트를 포함하는 표면 포인트들의 그리드의 1/4 에 속하는 표면 포인트에 대응하는 다른 각각의 제어 포인트 각각에 대해 엄격히 1 보다 더 높은 제 1 의 미리 결정된 수와 동

일하며, 다른 각각의 제어 포인트 각각에 대해 제 1의 미리 결정된 수의 10배보다 더 높은 제 2의 미리 결정된 수와 동일한 표시 함수이다;

[0020] - 이 방법은 그래픽 사용자-상호작용에 의해 제 1의 미리 결정된 수를 설정하는 단계를 포함한다;

[0021] - 프로그램은 업데이트 후 그리고 업데이트 전의 제어 포인트들 각각에 좌우되는 돌출 에너지이다;

[0022] - 프로그램은 타입
$$\sum_{i=0}^n \sum_{j=0}^m \mu_{ij} \|P_{ij} - Q_{ij}\|^2$$
 의 것이고 여기서 P_{ij} 및 Q_{ij} 는 업데이트 후 그리고 업데이트 전 각각의 제어 포인트들이다;

[0023] - 프로그램은 라그랑주 해결법 (Lagrange solving) 으로 해결된다;

[0024] - 라그랑주 해결법은, 입력으로서 선택된 표면 포인트들의 서브세트의 위치션을 취급하고 업데이트된 제어 포인트들을 출력하는 어픽스 함수 (affix function) 의 파라미터들을 결정하는 것, 그리고 그 후에 어픽스 함수를 선택된 표면 포인트들의 서브세트의 위치션에 수정된 대로 적용하는 것을 포함한다; 및/또는

[0025] - 선택된 표면 포인트들의 서브세트의 위치션을 수정하는 단계 및 제어 포인트들의 그리드를 업데이트하는 단계는 동일한 선택된 표면 포인트의 서브세트를 이용하여 반복되고, 어픽스 함수의 파라미터들은 초기 반복시 컴퓨팅 및 저장되고 그 후에 후속 반복들 전반에 걸쳐 취출된다.

[0026] 이 방법을 수행하기 위한 명령들을 포함하는 컴퓨터 프로그램이 또한 제공된다.

[0027] 컴퓨터 프로그램이 기록된 컴퓨터 판독가능 데이터 저장 매체가 또한 제공된다.

[0028] 그래픽 사용자 인터페이스 및 메모리에 커플링된 프로세서를 포함하는 CAD 시스템이 또한 제공되고, 메모리는 컴퓨터 프로그램을 기록한다.

[0029] 상기 설계 방법에 따라 산업 제품을 나타내는 3차원 오브젝트를 설계하는 단계들을 포함하고, 그 후에 설계된 3차원 오브젝트에 기초하여 산업 제품을 제조하는, 산업 제품을 제조하기 위한 방법이 또한 제공된다.

도면의 간단한 설명

[0030] 이제 본 발명의 실시형태들이 비제한적인 예에 의해, 그리고 첨부 도면들과 관련하여 설명될 것이다:

- 도 1 은 방법의 일 예의 플로우차트를 도시한 것이다;
- 도 2 는 시스템의 그래픽 사용자 인터페이스의 일 예를 도시한 것이다;
- 도 3 은 시스템의 일 예를 도시한 것이다; 그리고
- 도 4 내지 도 17 은 방법을 예시한 것이다.

발명을 실시하기 위한 구체적인 내용

[0031] 도 1 의 플로우차트를 참조하면, 3D 모델링된 오브젝트를 설계하기 위한 컴퓨터 구현 방법이 제안된다. 이 방법은 NURBS 표면 (예를 들어, Bézier 표면) 을 컴퓨터 시스템에 제공하는 단계 (S10) 를 포함한다. NURBS 는 3D 모델링된 오브젝트를 나타내고 (예를 들어, 3D 모델링된 오브젝트는 기계 부품과 같은 산업 제품, 예를 들어, 카메라의 외측 경계/엔벨로프의 적어도 일부를 -기하학적으로- 나타냄), 본질적으로 알려진 바와 같이, 제어 포인트들의 그리드에 의해 정의된다. 이 방법은 또한, 컴퓨터 시스템에 의해, NURBS 표면에 속하는 표면 포인트들의 그리드를 결정하는 단계 (S20) 를 포함한다. S20 에서 결정된 표면 포인트들의 그리드는 미리 결정된 가역 함수 (예를 들어, 파라미터들의 주어진 커플에서의 NURBS 표면의 평가) 에 따른 제어 포인트들의 그리드에 대응한다. 컴퓨터 시스템은 상기 미리 결정된 가역 함수, 또는 상기 미리 결정된 가역 함수가 유도될 수 있게 하는 룰들을 유지할 수도 있어서, 이 컴퓨터 시스템이 S20 을 언제든 수행할 수도 있다. 이것은 예컨대 CAD 의 분야에서 알려진 것이고, 여기서 개발자들은 이처럼 제공되었을 때 수학적 스킴들을 구현하는 방법을 안다 (수학적 스킴들의 예들은 추후에 제공되는 현황에 대해 적용가능함). 이 방법은 또한, 컴퓨터 시스템에 의해, NURBS 표면을 그리고, NURBS 표면 상에, 표면 포인트들의 그리드를 (예를 들어, 사용자에게, 예를 들어, 스크린 상에서) 디스플레이하는 단계 (S30) 를 포함한다. 이 방법은 그 후에 표면

포인트들의 서브세트 (예를 들어, 임의의 서브세트) 를 그래픽 사용자-상호작용에 의해 선택하는 단계 (S40) 를 포함한다. 이 방법은 그 후에 블록 S50 에서 그래픽 사용자-상호작용에 의해 선택된 표면 포인트들의 서브 세트의 포지션을 수정하는 단계 (S52) 를 수행한다. 이에 따라, 이 방법은 컴퓨터 시스템에 의해 (예를 들어, 자동으로) 표면을 실시간으로 업데이트하는 단계 (S54) 를 수행한다. 이 업데이트는 S52 에서 수행된 수정에 따라 (상기 수정에 대해 실시간으로 그리고 예를 들어 자동적 및/또는 즉각적 반응으로) 그리고 미리 결정된 가역 함수에 따라 수행된다. 도 1 에 도시된 바와 같이, 업데이트하는 단계 (S54) 는 NURBS 표면의 정의에 대해 반영되어서, 루프에서 S10, S20, S30 을 되풀이할 수도 있다. 이러한 점에서, S40 및/또는 S50 도 물론 다시 반복될 수도 있다. 이러한 루핑은 숙련된 컴퓨터 과학자에게 명백하다.

[0032] 이러한 방법은 3D 모델링된 오브젝트의 설계를 개선시킨다. 특히, 이 방법은, 그래픽 사용자-상호작용에 의해, NURBS 표면, CAD 및 산업 설계의 분야에서 널리 사용되는 타입의 표면의 수정을 허용한다. 그래픽 사용자-상호작용은 (S40 에서) 포인트들을 선택하는 것 그리고 그 후에 (S52 에서) 이러한 포인트들의 포지션을 수정하는 것으로 이루어지기 때문에 쉽다. S40 에서 선택되고 S52 에서 수정된 포인트들이 표면 포인트들이기 때문에, 사용자가 표면과 직접적으로 상호작용하고 로컬 조작으로 지오메트리를 수정할 수 있으므로, 이 방법은 매우 직관적인 그래픽 사용자-상호작용을 허용한다 (사용자가 NURBS 의 제어 포인트들을 조작해야 하는 솔루션들과 대조적임, 상기 제어 포인트들은 표면 상에 놓여있지 않음 - 이 방법의 조작되는 포인트들과는 다름). 이 방법은 그에 따라 "what you see is what you get" 거동을 제시한다. 또한, (예를 들어, 입력 수정의 진폭 및 국부성을 고려하는) 미리 결정된 가역 함수를 통해 NURBS 표면의 초기 제어 포인트들에 관련된 포인트들에 대해 수정이 수행되므로, 이 방법은 (S54 에서) 이러한 미리 결정된 가역 함수에 의존하여 표면 상에 수정을 반영하는 것이 가능하여, (특히 표면이 사용자에 의해 이전에 설계된 경우) 사용자 의도 및 표면의 초기 토폴로지를 유지한다. 실제로, 이 방법은 제어 포인트들을 삭제하지도 않고 제어 포인트들을 부가하지도 않는다. 이 방법은 단지 제어 포인트들을 수정하고, 표면을 정의하는 임의의 다른 데이터를 미터치된 채 남겨둘 수도 있다. 사용자 관점에서, 표면은 대체되지 않고 단지 수정될 뿐이다. 이것은 주어진 NURBS 표면의 본질적인 수학적 정의 및 상세 밀도를 보존한다. 그에 의해 사용자는 수정 및 결과적인 표면을 보다 양호하게 이해한다.

[0033] 또한, 시스템은, 예를 들어, 임의의 구성에서, S40 에서의 선택을 위한 예들에서 임의의 수의 포인트(들) 를 제공할 수도 있어서, 이 방법은 빠르고 유연한 수정 동작들을 허용한다. 특히, 예를 들어, 그리드의 동일한 로우/컬럼 상에 있지 않은, 예를 들어, 표면 포인트들의 그리드 상에서 반드시 위상적으로 인접하지는 않은 (즉, 그리드 상에서 반드시 이웃하지는 않은), 몇몇 표면 포인트들이 선택되고 수정될 수도 있다. 수정은, 예를 들어, 선택된 표면 포인트들 상에서 반드시 동종의 방법으로는 아닌, 임의의 방법으로 수행될 수도 있다. 예들에서, 이 방법, 특히 S54 는, 선행 시간에 수행되어서, 고도로 응답적이고 그에 따라 사용자 친화적이다. 예들에서, 이 방법은 수정에 의해 유도된 오실레이션들의 도입을 제한하여, 더욱 직관적인 결과를 제공한다.

[0034] 이 방법은 컴퓨터 구현된다. 이 방법의 단계들은 실제로 컴퓨터 시스템 상에서, 컴퓨터 시스템 단독으로 또는 그래픽 사용자-상호작용 (즉, 사용자가 컴퓨터 시스템의 GUI 와 상호작용함) 에 의해 실행된다. 따라서, 컴퓨터 시스템에 의해 특정적으로 수행된 단계들은 완전히 자동으로 수행될 수도 있다. 예들에서, 이 방법의 단계들 중 적어도 일부의 단계의 트리거링은 사용자-컴퓨터 상호작용을 통해 수행될 수도 있다. 특히, 제공하는 단계 (S10) 는 사용자에 의해 수행되거나 또는 트리거링될 수도 있다. 요구된 사용자-컴퓨터 상호작용의 레벨은 사용자의 원하는 것들을 구현할 필요성과 균형을 맞추고 예견된 자동기술법의 레벨에 좌우될 수도 있다. 예들에서, 이 레벨은 사용자 정의되거나 및/또는 미리 정의될 수도 있다.

[0035] 이 방법의 컴퓨터 구현의 통상적인 예는, 이 목적을 위해 적응된 시스템으로 이 방법을 수행하는 것이다. 시스템은 그래픽 사용자 인터페이스 (GUI) 및 메모리에 커플링된 프로세서를 포함할 수도 있고, 메모리는 이 방법을 수행하기 위한 명령들을 포함하는 컴퓨터 프로그램을 기록하고 있다. 메모리는 또한 데이터베이스를 저장할 수도 있다. 메모리는, 가능하다면 몇몇 물리적 개별 부품들 (예를 들어, 프로그램용으로 하나, 그리고 가능하다면 데이터베이스용으로 하나) 을 포함하는, 이러한 저장을 위해 적응된 임의의 하드웨어이다.

[0036] 이 방법은 일반적으로 모델링된 오브젝트들을 조작한다. 모델링된 오브젝트는 데이터베이스에 저장된 데이터에 의해 정의된 임의의 오브젝트이다. 확장에 의해, 표현 "모델링된 오브젝트" 는 데이터 그 자체를 나타낸다. "3D 모델링된 오브젝트를 설계하는 것" 은 3D 모델링된 오브젝트를 정교하게 하는 프로세스의 적어도 일부인 임의의 액션 또는 일련의 액션들을 나타낸다. 따라서, 이 방법은 새로이 3D 모델링된 오브젝트를 생성하는 것을 포함할 수도 있다. 대안적으로, 이 방법은 이전에 생성된 3D 모델링된 오브젝트를 제공하는

것, 그리고 그 후에 3D 모델링된 오브젝트를 수정하는 것을 포함할 수도 있다.

[0037] 시스템의 타입에 따라, 모델링된 오브젝트들은 상이한 종류들의 데이터에 의해 정의될 수도 있다. 시스템은 실제로 CAD 시스템, CAE 시스템, CAM 시스템, PDM 시스템 및/또는 PLM 시스템의 임의의 조합일 수도 있다. 이들 상이한 시스템들에서, 모델링된 오브젝트들은 대응하는 데이터에 의해 정의된다. 그것은 이에 따라 CAD 오브젝트, PLM 오브젝트, PDM 오브젝트, CAE 오브젝트, CAM 오브젝트, CAD 데이터, PLM 데이터, PDM 데이터, CAM 데이터, CAE 데이터라고 말할 수도 있다. 그러나, 모델링된 오브젝트가 이들 시스템들의 임의의 조합에 대응하는 데이터에 의해 정의될 수도 있으므로, 이들 시스템들은 다른 것에 배타적인 시스템이 아니다. 따라서, 시스템이 CAD 및 PLM 시스템 양쪽 모두일 수도 있는데, 이는 아래에 제공되는 이러한 시스템들의 정의들로부터 분명해질 것이다.

[0038] CAD 시스템은, CATIA 와 같은 모델링된 오브젝트의 그래픽 표현에 기초하여 모델링된 오브젝트를 설계하도록 적어도 적용된 임의의 시스템을 의미한다. 이 경우에, 모델링된 오브젝트를 정의하는 데이터는, 모델링된 오브젝트의 표현을 허용하는 데이터를 포함한다. CAD 시스템은, 예를 들어, 에지들 또는 라인들을 사용하여, 특정 경우들에서는 면들 또는 표면들로, CAD 모델링된 오브젝트들의 표현을 제공할 수도 있다. 라인들, 에지들, 또는 표면들은, 다양한 방식으로, 예를 들어, NURBS (non-uniform rational B-splines) 로 나타낼 수도 있다. 구체적으로는, CAD 파일은 명세 (specification) 들을 포함하고, 이로부터 지오메트리가 생성될 수도 있고, 이는 차례로 표현이 생성되게 한다. 모델링된 오브젝트의 명세들은 단일 CAD 파일 또는 다수의 파일들에 저장될 수도 있다. CAD 시스템에서 모델링된 오브젝트를 나타내는 파일의 통상적인 사이즈는 부품 당 1 메가바이트의 범위에 있다. 그리고, 모델링된 오브젝트는 통상적으로, 수천의 부품들의 어셈블리일 수도 있다.

[0039] CAD 시스템의 맥락에서, 모델링된 오브젝트는 통상적으로, 예를 들어, 부품 또는 부품들의 어셈블리와 같은 제품, 또는 가능하게는 제품들의 어셈블리를 나타내는, 3D 모델링된 오브젝트일 수도 있다. "3D 모델링된 오브젝트" 는, 그의 3D 표현을 허용하는 데이터에 의해 모델링되는 임의의 오브젝트를 의미한다. 3D 표현은 모든 각도들로부터 부품의 뷰잉을 허용한다. 예를 들어, 3D 모델링된 오브젝트는, 3D 로 나타낼 때, 그의 축들 중 임의의 축을 중심으로 또는 표현이 디스플레이되는 스크린 상에 있는 임의의 축을 중심으로 회전 및 핸들링될 수도 있다. 이것은 특히, 3D 모델링되지 않는 2D 아이콘을 제외한다. 3D 표현의 디스플레이는 설계를 용이하게 한다 (즉, 통계적으로 설계자가 그의 작업 (task) 을 달성하는 속도를 증가시킨다). 이것은 산업에서 제조 프로세스의 속도를 빠르게 하는데, 이는 제품들의 설계가 제조 또는 모델링 프로세스의 부분이기 때문이다.

[0040] 3D 모델링된 오브젝트는, 예를 들어, (예를 들어, 기계적) 부품 또는 부품들의 어셈블리, 또는 더욱 일반적으로는 임의의 강체 어셈블리 (예를 들어, 모바일 메카니즘) 와 같은, CAD 소프트웨어 솔루션 또는 CAD 시스템에 의한 그의 가상 설계의 완료에 후속하는 실제 세계에서 제조될 제품의 지오메트리를 나타낼 수도 있다. CAD 소프트웨어 솔루션은, 항공우주, 아키텍처, 건설, 소비 상품들, 하이-테크 디바이스들, 산업 장비, 운송, 해양, 및/또는 연안 또는 운송을 포함하는, 다양한 비제한의 산업 분야들에서 제품들의 설계를 허용한다. 이 방법에 의해 설계된 3D 모델링된 오브젝트는 그에 따라 지상 운반체 (예를 들어, 자동차 및 소형 트럭 장비, 경주용 자동차들, 모터사이클들, 트럭 및 모터 장비, 트럭들 및 버스들, 열차들을 포함함) 의 일부분, 공중 운반체 (예를 들어, 기체 장비, 항공우주 장비, 추진 장비, 방위 제품들, 항공기 장비, 우주 장비를 포함함) 의 일부분, 군함 운반체 (예를 들어, 해군 장비, 상선들, 연안 장비, 요트들 및 워크보트들, 해양 장비를 포함함) 의 일부분, 기계 부품 (예를 들어, 산업 제조 기계류, 중장비 모바일 기계류 또는 장비, 인스톨된 장비, 산업 장비 제품, 제조된 금속 제품, 타이어 제조 제품을 포함함), 전기-기계 또는 전자 부품 (예를 들어, 가전 제품들, 보안 및/또는 제어 및/또는 계측 제품들, 컴퓨팅 및 통신 장비, 반도체들, 의료 디바이스들 및 장비를 포함함), 소비 상품 (예를 들어, 가구, 가정 및 정원 제품들, 레저 상품들, 패션 제품들, 내구 상품 소매상의 제품들, 식물 상품 소매상의 제품들을 포함함), 패키징 (예를 들어, 음식 및 음료 및 담배, 미용 및 개인 위생, 가정용품 패키징을 포함함) 일 수도 있는 산업 제품을 나타낸다. 이 방법은 3D 모델링된 오브젝트의 미적 양태들이 높은 중요성을 갖는 경우에 특히 유용하다. 이것은 3D 모델링된 오브젝트가 운반체 또는 비행기 기계 부품, 예컨대 가시적 부품, 또는 소비 상품 (예를 들어, 카메라와 같은 전자 상품) 을 나타낼 때 해당된다.

[0041] PLM 시스템은, 물리적 제조 제품을 나타내는 모델링된 오브젝트의 관리를 위해 적용된 임의의 시스템을 의미한다. PLM 시스템에서, 모델링된 오브젝트는 그에 따라 물리적 오브젝트의 제조에 적합한 데이터에 의해 정의된다. 이들은 통상적으로 디멘전 값들 및/또는 공차 (tolerance) 값들일 수도 있다. 오브젝트의 올바른

제조를 위해, 실제로 이러한 값들을 갖는 것이 더 좋다.

[0042] CAM 은 컴퓨터 지원 제조 (Computer-Aided Manufacturing) 를 나타낸다. CAM 솔루션은, 제품의 제조 데이터를 관리하도록 적용된 임의의 솔루션, 하드웨어의 소프트웨어를 의미한다. 제조 데이터는 일반적으로, 제조할 제품, 제조 프로세스 및 요구된 리소스들에 관련된 데이터를 포함한다. CAM 솔루션은 제품의 전체 제조 프로세스를 계획하고 최적화하는데 사용된다. 예를 들어, 그것은 CAM 사용자들에게 실현가능성에 대한 정보, 제조 프로세스의 지속기간, 또는 제조 프로세스의 특정 단계에서 사용될 수도 있는 특정 로봇들과 같은 리소스들의 수를 제공할 수 있고; 따라서 관리 또는 필요한 투자에 대한 관정을 허용한다. CAM 은, CAD 프로세스 및 잠재적인 CAE 프로세스 후의 후속 프로세스이다. 이러한 CAM 솔루션들은, 상표 DELMIA® 로 Dassault Systèmes 에 의해 제공된다.

[0043] CAE 는 컴퓨터 지원 엔지니어링 (Computer-Aided Engineering) 을 나타낸다. CAE 솔루션은, 모델링된 오브젝트의 물리적 거동의 분석을 위해 적용된 임의의 솔루션, 하드웨어의 소프트웨어를 의미한다. 잘 알려져 있고 널리 사용되는 CAE 기법은 FEM (Finite Element Method) 이고, 이는 모델링된 오브젝트를 엘리먼트들로 분할하는 것을 통상적으로 수반하고 그 물리적인 거동들은 방정식들을 통해 컴퓨팅 및 시뮬레이션될 수 있다. 이러한 CAE 솔루션들은, 상표 SIMULIA® 로 Dassault Systèmes 에 의해 제공된다. 다른 성장하는 CAE 기법은, CAD 지오메트리 데이터 없이 물리 (physics) 의 상이한 필드들로부터 복수 컴포넌트들로 구성되는 복잡한 시스템들의 모델링 및 분석을 수반한다. CAE 솔루션들은, 제조할 제품들의 시뮬레이션 그리고 따라서 최적화, 개선 및 입증 (validation) 을 허용한다. 이러한 CAE 솔루션들은, 상표 DYMOLA® 로 Dassault Systèmes 에 의해 제공된다.

[0044] PDM 은 제품 데이터 관리 (Product Data Management) 를 나타낸다. PDM 솔루션은, 특정 제품에 관한 모든 타입의 데이터를 관리하도록 적용된 임의의 솔루션, 하드웨어의 소프트웨어를 의미한다. PDM 솔루션은, 제품의 라이프사이클에 수반되는 모든 행위자들: 주로 엔지니어들이지만 또한 프로젝트 관리자들, 재무원, 판매원 및 바이어들에 의해 사용될 수도 있다. PDM 솔루션은 일반적으로 제품 지향 데이터베이스에 기초한다. 그것은 사용자들로 하여금 그들의 제품들에 대한 일관성있는 데이터를 공유하게 하여서 행위자들이 불일치하는 데이터 (divergent data) 를 사용하는 것을 방지한다. 이러한 PDM 솔루션들은, 상표 ENOVIA® 로 Dassault Systèmes 에 의해 제공된다.

[0045] 도 2 는 시스템의 GUI 의 일 예를 도시하고, 여기서 시스템은 CAD 시스템이다.

[0046] GUI (2100) 는, 표준 메뉴 바들 (2110, 2120) 및 하부 및 측부 툴바들 (2140, 2150) 을 갖는 통상적인 CAD 형 인터페이스일 수도 있다. 이러한 메뉴 및 툴바들은 사용자 선택가능 아이콘들의 세트를 포함하고, 각 아이콘은, 업계에 알려진 바처럼, 하나 이상의 동작들 또는 기능들과 연관된다. 이들 아이콘들 중 일부는 GUI (2100) 에서 디스플레이되는 3D 모델링된 오브젝트 (2000) 상에서 편집 및/또는 작업하도록 적용된 소프트웨어 툴들과 연관되고, 디스플레이된 3D 모델링된 오브젝트 (2000) 는 예를 들어 이 방법을 수행한 결과이다. 소프트웨어 툴들은 워크벤치 (workbench) 들로 그룹화될 수도 있다. 각 워크벤치는 소프트웨어 툴들의 서브세트를 포함한다. 특히, 워크벤치들 중 하나는 모델링된 제품 (2000) 의 기하학적 피쳐들을 편집하는데 적합한 편집 워크벤치이다. 동작시, 설계자는, 예를 들어, 오브젝트 (2000) 의 일부분을 미리 선택하고 그 후에 동작 (예를 들어, 디멘션, 컬러 등의 변경) 을 개시하거나, 또는 적절한 아이콘을 선택하는 것에 의해 기하학적 제약들을 편집할 수도 있다. 예를 들어, 통상적인 CAD 동작들은 스크린 상에 디스플레이된 3D 모델링된 오브젝트의 폴딩 (folding) 또는 펀칭 (punching) 의 모델링이다.

[0047] GUI 는, 예를 들어, 디스플레이된 제품 (2000) 에 관련된 데이터 (2500) 를 디스플레이할 수도 있다. 도 2 의 예에서, "피쳐 트리" 로서 디스플레이된 데이터 (2500) 및 그들의 3D 표현 (2000) 은 브레이크 캘리퍼 및 디스크 등을 포함하는 브레이크 어셈블리에 속한다. GUI 는 또한, 예를 들어, 오브젝트의 3D 오리엔테이션을 용이하게 하는 것, 디스플레이된 제품 (2000) 의 다양한 속성들을 렌더링하거나 또는 편집된 제품의 동작의 시뮬레이션을 트리거링하는 것을 위한, 다양한 타입들의 그래픽 툴들 (2130, 2070, 2080) 을 보여줄 수도 있다. 커서 (2060) 는 사용자로 하여금 그래픽 툴들과 상호작용하게 하기 위해 햅틱 디바이스에 의해 제어될 수도 있다.

[0048] 도 3 은 시스템의 일 예를 도시하고, 여기서 시스템은 클라이언트 컴퓨터 시스템, 예를 들어, 사용자의 워크스테이션이다.

[0049] 이 예의 클라이언트 컴퓨터는 내부 통신 버스 (1000) 에 연결된 중앙 프로세싱 유닛 (CPU; 1010), 버스에 또한 연결된 랜덤 액세스 메모리 (RAM; 1070) 를 포함한다. 클라이언트 컴퓨터에는 버스에 연결된 비디오 랜덤 액세스 메모리 (1100) 와 연관되는 그래픽 프로세싱 유닛 (GPU; 1110) 이 추가로 제공된다. 비디오 RAM (1100) 은 또한 프레임 버퍼로서 업계에 알려져 있다. 대용량 저장 디바이스 제어기 (1020) 는 대용량 메모리 디바이스, 예컨대, 하드 드라이브 (1030) 에의 액세스들을 관리한다. 컴퓨터 프로그램 명령들 및 데이터를 유형적으로 구현하는데 적합한 대용량 메모리 디바이스들은, 예로서, EPROM, EEPROM, 및 플래시 메모리 디바이스들과 같은 반도체 메모리 디바이스들; 내부 하드 디스크들 및 착탈식 디스크들과 같은 자기 디스크들; 광자기 디스크들; 및 CD-ROM 디스크들 (1040) 을 포함한, 모든 형태들의 비휘발성 메모리를 포함한다. 전술한 것들 중 임의의 것이 특별히 설계된 ASIC들 (주문형 집적 회로들) 에 의해 보충되거나 또는 그 안에 통합될 수도 있다. 네트워크 어댑터 (1050) 는 네트워크 (1060) 에의 액세스들을 관리한다. 클라이언트 컴퓨터는 또한, 커서 제어 디바이스, 키보드 등과 같은 햅틱 디바이스 (1090) 를 포함할 수도 있다. 커서 제어 디바이스는 사용자로 하여금 디스플레이 (1080) 상의 임의의 원하는 로케이션에 커서를 선택적으로 포지셔닝시킬 수 있도록 하기 위해 클라이언트 컴퓨터에 사용된다. 또한, 커서 제어 디바이스는 사용자로 하여금 다양한 커맨드들을 선택하게 하고, 제어 신호들을 입력하게 한다. 커서 제어 디바이스는 시스템으로의 입력 제어 신호들을 위한 다수의 신호 생성 디바이스들을 포함한다. 통상적으로, 커서 제어 디바이스는 마우스일 수도 있으며, 이 마우스의 버튼은 신호들을 생성하는데 사용된다. 대안적으로 또는 부가적으로, 클라이언트 컴퓨터 시스템은 감응형/터치 패드, 및/또는 감응형/터치 스크린을 포함할 수도 있다.

[0050] 컴퓨터 프로그램은 컴퓨터에 의해 실행가능한 명령들을 포함할 수도 있고, 그 명령들은 위의 시스템으로 하여금 방법을 수행하게 하는 수단을 포함한다. 프로그램은 시스템의 메모리를 포함하는, 임의의 데이터 저장 매체 상에 기록가능할 수도 있다. 프로그램은, 예를 들어, 디지털 전자 회로부에서, 또는 컴퓨터 하드웨어, 펌웨어, 소프트웨어에서, 또는 이들의 조합에서 구현될 수도 있다. 프로그램은 장치로서, 예를 들어, 프로그램 가능 프로세서에 의한 실행을 위해 머신 판독가능 저장 디바이스에서 유형적으로 구현된 제품으로서 구현될 수도 있다. 방법 단계들은 입력 데이터 상에서 동작하고 출력을 생성함으로써 방법의 기능들을 수행하기 위해 명령들의 프로그램을 실행하는 프로그램가능 프로세서에 의해 수행될 수도 있다. 프로세서는 따라서 프로그램가능하고, 데이터 저장 시스템, 적어도 하나의 입력 디바이스, 및 적어도 하나의 출력 디바이스로부터 데이터 및 명령들을 수신하고, 이들로 데이터를 및 명령들을 송신하도록 커플링될 수도 있다. 애플리케이션 프로그램은 고-레벨의 절차적 또는 오브젝트-지향된 프로그래밍 언어로, 또는 원한다면 어셈블리 또는 머신 언어로 구현될 수도 있다. 어느 경우든, 이 언어는 컴파일링 또는 해석된 언어일 수도 있다. 프로그램은 전체 설치 (full installation) 프로그램 또는 업데이트 프로그램일 수도 있다. 시스템 상에서의 프로그램의 애플리케이션은 어느 경우든 방법을 수행하기 위한 명령들을 발생시킨다.

[0051] 표면 포인트들의 그리드가 소위 "Gréville" (또는 "Greville") 포인트들의 그리드인 방법의 일 예가 이하 설명된다. Gréville 포인트들은 Gréville 가로축 (abscissae) 에서 NURBS 표면의 평가에 의해 주어진 3D 포인트들이다. 또한, Gréville 가로축은 NURBS 표면의 제어 포인트들의 파라미터들이다. NURBS 표면이 S10 이전에 전문 설계자에 의해 설계된다는 통상의 시나리오에서, Gréville 포인트들은 동일한 정도의 길이 (예를 들어, 세그먼트들의 적어도 80% 에 대해, 예를 들어, 평균 길이에 대해 플러스 또는 마이너스 20%) 를 갖는 - 그리드 상의 연속적인 Gréville 포인트들 사이에서의 - 세그먼트들을 갖는 동종 다면체를 형성한다. 따라서, Gréville 포인트들은 S10 에서 제공된 표면의 초기 설계자의 사용자-의도를 전달하여, 이들 포인트들의 조작은 특히 이 방법을 특히 사용자 친화적으로 만든다. Gréville 포인트들의 그리드는 초기 설계자가 모델에 추가하기를 원했던 상세들의 로컬 밀도들 및 제어 포인트들의 그리드를 반영한다. 실제로, 설계자가 상세들을 추가하기를 위한 경우, Gréville 포인트들은 설계자가 이러한 상세들을 원하지 않은 경우에 비해 더 높은 로컬 밀도를 갖는다. 이 방법은 따라서 S10 에서 제공된 표면이 S10 이전에 휴먼 설계자에 의해 설계되는 글로벌 프로세스에서 특히 유용하다는 것을 입증할 수도 있고, 로컬 존들이 몇몇 스케일로 로컬 형상들을 제시한다. Gréville 포인트들의 밀도는 존의 표면 밀도에 의해 분할된 존의 Gréville 포인트들의 수로서 확인될 수도 있고, 이러한 경우 로컬 존들이 고레벨 상세들을 제시하는 경우에 (통계적으로) 더 높을 것이다. Gréville 포인트들의 그리드가 NURBS 표면의 동-가로축 상에 정의되므로, 경계 효과들이 제한된다.

[0052] 이 방법은 3D 모델링된 오브젝트 (예를 들어, 그 경계) 를 나타내는 NURBS 표면의 수정을 허용한다. 본질적

으로 알고 있는 바와 같이, NURBS 표면은, 예를 들어, 몇몇 (적어도 2개) 로우들 및 몇몇 (적어도 2개) 컬럼들을 갖는 "그리드" 의 통상적인 위상적 정의를 이용하여, 소위 "제어 포인트들" 의 (2D) 그리드에 의해 정의된다. 이 방법 전반에 걸쳐, 컴퓨터 시스템은 NURBS 표면에 속하고 미리 결정된 가역 함수에 따른 제어 포인트들의 그리드에 대응하는 표면 포인트들의 그리드를 결정한다 (S20). 표면 포인트들은 기하학적으로 표면 상에 있는/표면에 속하는 포인트들이다. NURBS 의 경우, 표면의 직통적 (straightforward) 그리드는 이 예의 Gréville 포인트들의 그리드이다. NURBS 의 표면 Gréville 포인트들과 제어 포인트들 (일반적으로 표면 자신의 외측) 사이의 일대일 대응을 허용하는 미리 결정된 가역 함수는 미리 결정되고, NURBS 지오메트리를 설명하는 널리 알려진 수학적 방법들로부터 제공된다. 그러나, 이들 수학적 방법들은 완벽을 기하기 위해 아래에 상세화된다. 결정하는 단계 (S20) 는 임의의 방법으로, 예를 들어, 표면 및 그의 제어 포인트들이 수정될 때마다 Gréville 포인트들을 실시간으로 재컴퓨팅하는 백그라운드 프로세스로서, 수행될 수도 있다는 것에 주목한다.

[0053] NURBS 표면은, 3D 공간에서의 값들을 갖는 2D 공간에서 정의된, CAD 에 널리 사용되는 표면이다: $S(u, v) \rightarrow (x, y, z)$. NURBS 표면은 제어 포인트들의 2차원 어레이 및 2개의 매듭 벡터들, 그리고 가능하면 사용된 CAD 소프트웨어에 좌우되는 다른 데이터로 나타낸다. 통상적으로, 매듭 벡터들은 부동 소수점 값들의 1D 어레이들이다. 제어 포인트들의 어레이는 인덱싱될 수도 있다. 각각의 제어 포인트는 좌표들 x, y, z 를 갖는 3D 포인트이다. 어레이에서의 위치는 2개의 인덱스들 i 및 j 로 정의된다; 여기서 $0 \leq i < M$ 그리고 $0 \leq j < N$ 이다. Gréville 파라미터들로서 알려진 파라미터들의 세트는 매듭 벡터들로부터 컴퓨팅될 수 있어서, 인덱스들 i 또는 j 의 각각의 값이 Gréville 파라미터 (u_i, v_j) 와 연관될 수 있다. 전통적으로, Gréville 파라미터들은 매듭 벡터의 일정 수의 연속적인 값들의 평균의 커플이다. "차수" 라고 불리는 이 일정 수는 매듭 벡터의 각각의 차수 (u 또는 v) 에 대해 상이할 수도 있다. 일반적으로, 차수들의 커플은 NURBS 표면을 나타내는 데이터에 속한다. 더욱 상세하게는, Gerald Farin 에 의한 논문 "Curves and surfaces for Computer Aided Geometric Design", ed. Morgan Kaufmann, (2001) 은 NURBS 표면들의 기초들을 제공한다.

[0054] 고전적으로, NURBS 표면은 다음 입력들에 의해 정의된다.

[0055] 1. $(n + 1) (m + 1)$ 제어 포인트들 $P_{ij} \in \mathbb{R}^3$ 의 그리드, 여기서 $i \in \{0, \dots, n\}$ 그리고 $j \in \{0, \dots, m\}$.

[0056] 2. 추후 설명되는 몇가지 이유들로 "차수들" 이라고 명명된 2개의 포지티브 정수들 p, q .

[0057] 3. 첫 번째 $p + 1$ 값들 u_i 가 동일하도록 그리고 마지막 $p + 1$ 값들 u_i 가 동일하도록 " u 매듭 벡터" 라고 명명된 실수들의 제 1 순서 리스트 $u_0 \leq \dots \leq u_i \leq u_{i+1} \leq \dots \leq u_{p+n+1}$.

[0058] 4. 첫 번째 $q + 1$ 값들 v_j 가 동일하도록 그리고 마지막 $q + 1$ 값들 v_j 가 동일하도록 " v 매듭 벡터" 라고 명명된 실수들의 제 2 순서 리스트 $v_0 \leq \dots \leq v_j \leq v_{j+1} \leq \dots \leq v_{q+m+1}$.

[0059] 5. $(n + 1) (m + 1)$ 가중치 수들 $\omega_{ij} > 0$ 의 그리드, 여기서 $i \in \{0, \dots, n\}$ 그리고 $j \in \{0, \dots, m\}$.

[0060] 이들 입력들은, $N_i^p: [u_0, u_{p+n+1}] \rightarrow \mathbb{R}^+$ 여기서 $i \in \{0, \dots, n\}$ 그리고 $N_j^q: [v_0, v_{q+m+1}] \rightarrow \mathbb{R}^+$ 여기서 $j \in \{0, \dots, m\}$ 라고 각각 쓰여지는 2개의 B-스플라인 기저를 정의한다. 기저 B-스플라인 함수들 N_i^p 및 N_j^q 는 각각의 차수들 p 및 q 를 갖는 구간적 다항식이다. 이들의 상세화된 정의는 본 발명에 유

용하지 않다. NURBS 표면 $S: [u_0, u_{p+n+1}] \times [v_0, v_{q+m+1}] \rightarrow \mathbb{R}^3$ 은 다음의 유리 파라미터화에 의해 정의된다.

$$S(u, v) = \frac{1}{\sum_{i=0}^n \sum_{j=0}^m \omega_{ij} N_i^p(u) N_j^q(v)} \sum_{i=0}^n \sum_{j=0}^m \omega_{ij} P_{ij} N_i^p(u) N_j^q(v)$$

이와 유사하게, NURBS 곡선들 $C: [t_0, t_{p+n+1}] \rightarrow \mathbb{R}^3$ 은 다음 식에 수반된 가중치들 ω_i 및 $n+1$ 제어 포인트들 P_i 의 리스트의 차수 p 에 의해 정의된다.

$$C(t) = \frac{1}{\sum_{i=0}^n \omega_i N_i^p(t)} \sum_{i=0}^n \omega_i P_i N_i^p(t)$$

실제로, 수학적 관점에서, NURBS 표면은 2개의 NURBS 곡선들의 텐서 곱이다. 명확성을 위해, 이 방법은 주로 (다음 도면들에서) NURBS 표면들보다는 NURBS 곡선들로 예시된다.

Gréville 포인트들이 이제 설명된다.

Gréville 가로축들은 (u_k^*, v_l^*) 로 쓰여지는 (u, v) 파라미터들의 특정 값들이고 여기서 $k \in \{0, \dots, n\}$ 이고 $l \in \{0, \dots, m\}$ 이다. 이들은 다음 식들에 의해 고유하게 정의된다.

$$u_k^* = \frac{1}{p} \sum_{\sigma=k}^{k+p-1} u_{\sigma} \quad v_l^* = \frac{1}{q} \sum_{\sigma=l}^{l+q-1} v_{\sigma}$$

다시 말해서, Gréville 가로축은 매듭 벡터의 p (resp. q) 연속적인 값들의 평균 값이다. 제어 포인트 P_{kl} 에 의해 가장 영향받은 표면 상의 포인트는 소위 Gréville 포인트 $S(u_k^*, v_l^*)$ 이다. 일부 방법으로, Gréville 포인트들은 표면 상의 제어 포인트들의 영향을 개시한다. Gréville 포인트들을 $G_{kl} = S(u_k^*, v_l^*)$ 로 쓴다면, 그리고 NURBS 표면의 정의에 따르면, Gréville 포인트들과 제어 포인트들 사이의 관계는 다음의 선형 식에 의해 정의된다 (그리고 이것은 "미리 결정된 가역 함수" 의 가능한 형태이다).

$$G_{kl} = \sum_{i=0}^n \sum_{j=0}^m a_{ijkl} P_{ij} \quad (k, l) \in \{0, \dots, n\} \times \{0, \dots, m\}$$

여기서

$$a_{ijkl} = \frac{\omega_{ij} N_i^p(u_k^*) N_j^q(v_l^*)}{\sum_{I=0}^n \sum_{J=0}^m \omega_{IJ} N_I^p(u_k^*) N_J^q(v_l^*)}$$

모든 Gréville 포인트들의 모든 좌표들의 로우 벡터를 G 로:

$$G = (G_{kl})_{(k,l) \in \{0, \dots, n\} \times \{0, \dots, m\}} \in \mathbb{R}^{3(n+1)(m+1)}$$

[0074] 그리고 모든 제어 포인트들의 모든 좌표들의 로우 벡터를 P 로:

$$P = (P_{ij})_{(i,j) \in \{0, \dots, n\} \times \{0, \dots, m\}} \in \mathbb{R}^{3(n+1)(m+1)}$$

[0075] 쓰는 것에 의해 더욱 알맞은 식이 획득된다.

[0076] 그 후에, 이전 식은 선형 변환 $A: \mathbb{R}^{3(n+1)(m+1)} \rightarrow \mathbb{R}^{3(n+1)(m+1)}$ 을 정의하여

$$G = AP$$

[0077] 이고

$$P = A^{-1}G$$

[0078] 이다.

[0079] 선형 변환 A 는 정방 $(3(n+1)(m+1)) \times (3(n+1)(m+1))$ 행렬이고, 종래 기술에 따르면, 이것은 가역가능하다 (모든 Gréville 포인트가 한 번 선택되는 경우 - 행렬에서의 주어진 로우가 두 번 나타나지 않는 경우).

[0080] 도 4 는 NURBS 곡선과 함께 그의 $n+1=6$ 제어 포인트들 P_i 및 Gréville 포인트들 G_i 를 예시한다.

[0081] 이제 이 방법은 사용자-상호작용을 통해 표면의 수정을 허용한다. 이를 위해, 컴퓨터 시스템은 NURBS 표면을 그리고, NURBS 표면 상에, 표면 포인트들의 그리드를 디스플레이한다 (S30). 디스플레이하는 단계 (S30) 는 여기서는 백그라운드 프로세스로서, 예를 들어, 전체 방법 전반에 걸쳐 (예를 들어, 수정들을 수행하기 위해, 예를 들어, 이 방법의 반복들 전반에 걸쳐) 다시 수행될 수도 있다. 디스플레이하는 단계 (S30) 는 CAD 에서 표면들을 디스플레이하는 임의의 고전적 방법으로, 예를 들어, 표면들의 경계들을 하이라이트하고 표면들을 셰이딩함으로써 수행될 수도 있다. 포인트들의 그리드는 컴퓨터 시스템의 디스플레이 디바이스 (예를 들어, 스크린) 상에서 블릿들 또는 정방형들로서 통상적으로 디스플레이되고 - 예를 들어, 예컨대 볼드로 하이라이트되고 -, 이들은 표면의 그래픽 표현 상에 놓여 디스플레이된다. 사용자에게 표면의 시각적 표현 및 표면 상의 표면 포인트들 (이 예에서 Gréville 포인트들) 의 그리드가 제공되는 한 임의의 다른 그래픽 표현이 사용될 수도 있다.

[0082] 사용자는 그 후에 S40 에서, 그래픽 사용자-상호작용에 의해, 표면 포인트들의 서브세트를 선택할 수도 있다. 그래픽 사용자-상호작용은 임의의 방법으로 수행될 수도 있다. 통상적으로, 사용자는 선택 기능을 론칭하고 그 후에 디스플레이된 Gréville 포인트들의 표현들과, 예를 들어, 커서, 터치펜, 또는 그의/그녀의 손가락(들) 과 상호작용한다. 대안적으로, 사용자는 (기능을 론칭하는 일 없이) Gréville 포인트들의 디스플레이된 그래픽 표현들과 직접적으로 상호작용할 수도 있다. 앞서 언급된 바와 같이, 컴퓨터 시스템은, 그리드의 동일한 로우 또는 컬럼 상에서가 아닌 몇몇 표면 포인트들을 포함하여, 표면 포인트들의 임의의 서브세트를 실질적으로 동시에 S40 에서의 선택을 위해 제공할 수도 있다. 실제로, 사용자는 레퍼런스 지오메트리로부터 동일한 거리를 갖는 몇몇 포인트들을 이동시키는 것을 원할 수도 있고, 이 방법은 그에 의해 단일 동작에서 이 액션의 달성을 위해 제공할 수도 있다. 사용자는, 예를 들어, 표면 포인트들을 번갈아 고를 수도 있다.

[0083] 사용자는 그 후에 S52 에서, 그래픽 사용자-상호작용에 의해, 선택된 표면 포인트들의 서브세트의 포지션을 수정할 수 있다. 이것은 임의의 방법으로, 예를 들어 드래그-앤드-드롭 동작에 의해 수행될 수도 있다. 드래그-앤드-드롭은 연속적으로 선택 S40 에 뒤따를 수도 있다. 예를 들어, 사용자는 디스플레이된 표면 포인트(들) 를 클릭, 터치 또는 더블 클릭 또는 터블 터치할 수 있다. 그 후에 사용자는 전체 선택의 클릭하는/터치하는 드래그-앤드-드롭을 수행할 수 있다. 시스템은 이러한 동작을 표면 포인트들의 포지션의 수정으로서 해석하고, 이러한 해석은 임의의 고전적인 방법으로 수행된다.

[0084] 컴퓨터 시스템과의 이러한 전체 그래픽 사용자-상호작용에 따르면, 이 방법은, 예를 들어, 컴퓨터 시스템에 의해 자동으로 표면을 실시간으로 업데이트한다 (S54). 이 업데이트 (S54) 는 미리 결정된 가역 함수, 즉,

Gréville 포인트들의 포지션과 NURBS 의 정의, 즉, 제어 포인트들의 포지션 사이의 수학적 링크를 유지하는 것에 따라 수행된다. NURBS 를 정의하는 모든 다른 상술된 데이터가 미터치된 채 남겨질 수도 있어서, (앞서 설명된 바와 같이) 사용자-의도를 보존한다. 업데이트 (S54) 의 구현은 상기 제공된 수학적 설명들에서 이어진다. 업데이트 (S54) 는 그에 의해 (제어 포인트들의 포지션을 수정함으로써) 표면 그리고 그에 따라 디스플레이 (S30) 를 수정하는 동안, NURBS 표면의 초기 구조 (매듭 벡터들에 대해서는 전혀 수정되지 않음) 그리고 그에 따라 사용자-의도를 유지한다. 이 동작은 사용자가 원하는 만큼 여러번 반복될 수도 있는데, 이 반복들이 어떻게 핸들링되는지에 주목하고, 수정들을 입증 및/또는 취소하는 문제들이 구현 상세들이다.

[0088] 전통적인 모델들의 결점은 제어 포인트들이 표면 상에 위치되지 않는다는 점이다. 이들은 표면으로부터 멀리 있는 그리드를 정의하여, 표면의 편집은 간접적인 프로세스이다: 제어 포인트를 이동시키는 것, 표면의 변경을 체크하는 것, 필요하다면, 제어 포인트를 다시 이동시키는 것 등. 물리 기반 방법들은 힘 인가를 수반하고 동일한 결점을 경험하는데 이는 사용자가 힘의 중심지 및 크기를 조정할 필요가 있기 때문이고, 이는, 실제로, 간접적인 조작이다. 힘을 인가함으로써 표면을 변형시키는 것은 때때로 "직접적인 조작" 이라고 명명되지만, 이것은 사용자가 표면 상의 포인트들을 실제로는 다루지 않기 때문에 부적절하다는 것에 주목해야 한다. 또한, 물리 기반 방법들은 비선형 미분 방정식 해결법 및 과잉 결정된 충돌 제약들을 수반하고, 이는 컴퓨팅 시간 소비적일 수도 있다. 제어 포인트들은 CAD 산업에 널리 사용되지만, 직접적인 편집 프로세스를 통해 표면 상에 위치된 포인트들을 이동시킴으로써 표면의 형상을 변경시키는 능력이 유리하다.

[0089] 이 예의 방법은, 본래부터 표면 상에 위치된 Gréville 포인트들을 사용함으로써 NURBS 표면들의 직접적인 변형을 허용한다. 또한, 평활화 프로세스를 통해 그리고 이웃 지오메트리를 수반함으로써, 추후에 설명되는 다른 예에서의 방법은 Gréville 포인트들 편집의 원하지 않는 영향들을 피하게 한다. 종래 기술과는 대조적으로, 이 방법은, 표면 상의 포인트들을 고르는 것에 의한, 그리고 일부 외부 지오메트리를 고르지 않는 것에 의한, 직접적인 변형을 허용한다. 이러한 직접적인 변형은 평활화되고 완만하게 되어 매우 대중적인 제어 포인트들의 직관적인 거동이 보존된다. 수학적 해결법은 선형 방정식들에 기초하여, 추후에 제공되는 예들에서처럼, 기하학적 업데이트를 매우 빠르고 신뢰성있게 만든다. 이것은 스타일링 설계 생산성을 개선시킨다.

[0090] 직통적 솔루션은 사용자가 Gréville 포인트들 G 의 포지션들을 G' 로 변경시키고 $P' = A^{-1}G'$ 를 사용함으로써 새로운 제어 포인트들 P' 를 컴퓨팅하는 것이다. 본 발명자들에 의해 수행된 실험들은, 도 5 에 예시된 바와 같이, 원하지 않는 오실레이션들이 이 솔루션에 의해 표면 상에 생성된다는 것을 보여준다. 좌측 부분은 6개의 제어 포인트들 (예시되지 않음) 에 의해 정의된 수평 NURBS 곡선이다. 도트들은 곡선의 Gréville 포인트들이다. Gréville 포인트 G_4 는 G'_4 로 이동된다. 우측 부분은 예기치 않은 오실레이션들을 특징으로 하는 결과적인 NURBS 곡선을 예시한다. 도트로 된 폴리라인은 NURBS 곡선의 제어 다각형이다. 이러한 부작용은 Gréville 포인트들이 전통적으로 NURBS 표면들을 조작하는데 사용되지 않는 이유이다. 이 문제는 다음의 잘못된 구성으로부터 온다. 실생활에서, 사용자는 모든 Gréville 포인트들을 동시에 변경하지 않지만, 식 $P' = A^{-1}G'$ 는 요청을 바꾼다: 사용자에게 의해 선택된 Gréville 포인트들을 고려하고 모든 다른 Gréville 포인트들을 미변경된 채로 유지한다. 선택되지 않은 Gréville 포인트들을 미변경된 채로 유지하는 것은 이 솔루션을 곤란하게 만든다.

[0091] 이 방법은 소위 "완화" 를 수행함으로써 이러한 이슈를 경감시킬 수도 있다. 그 생각은 사용자에게 의해 선택되지 않은 Gréville 포인트들을 자유롭게 하기 위한 것이다. 이러한 자유도 덕분에, 선택되지 않은 제어 포인트들이 대체되어 사용자의 예기된 평활화 거동을 유발한다. 구체적으로, 제어 포인트들의 그리드를 업데이트하는 단계 (S54) 는, 선택된 표면 포인트들의 서브세트의 포지션이 수정된 대로 (즉, S52 에서 사용자에게 의해 요청된 대로) 있다는 (예를 들어, 고유한) 제약 하에서, NURBS 표면의 변형에 제재를 가하는 프로그램을 최적화함으로써 수행될 수도 있다. 다시 말해서, 업데이트하는 단계 (S54) 는 - 필요하다면 선택되지 않은 Gréville 포인트들을 이동시킴으로써 - NURBS 의 지오메트리의 전체 변형을 최소화하면서, 사용자에게 의해 선택되고 이동된 Gréville 포인트들이 결국 사용자가 이들을 이동시켰던 것이라는 것을 단지 보장한다 (일반적으로 말하면, 적용된 수학적 방법들의 분야로부터 알려진 바와 같이, 그 목표로 향하는 프로그램이 적절한 제재 조항

들로 최적화됨). 따라서, 선택되지 않은 Gréville 포인트들은 S54 를 수행할 때 자유롭게 되거나, 또는 "완화된다".

[0092] NURBS 표면의 변형을 제재를 가하는 것으로 유지된 기준은 이러한 기능을 이행하는 임의의 미리 결정된 수학적 기준이다. 일 예에서, 프로그램은, 각각의 제어 포인트 각각에 대해, 업데이트 전의 각각의 제어 포인트와 업데이트 후의 각각의 제어 포인트 사이의 거리에 제재를 가한다. 임의의 거리, 예를 들어, 유클리드 거리 (또는 솔루션의 단일성을 보장하는, 돌출된 에너지를 유도하는 임의의 이러한 거리) 가 사용될 수도 있다. 다시 말해서, 프로그램은 (물론 사용자에게 의해 부과된 제약을 고려하면서) 가능한 많이, 업데이트 (S54) 후에 제어 포인트들을 이들의 오리지널 포지션을 향해 "압하한다". 이것은 빠른 컴퓨테이션들을 허용한다.

[0093] Q_{ij} 가 사용자가 변형을 시작하기 전의 표면인 초기 표면의 제어 포인트들이라고 하자. $\tilde{G} = (G_{kl})_{(k,l) \in V}$ (여기서 $V \subset \{0, \dots, n\} \times \{0, \dots, m\}$) 가 사용자에게 의해 선택된 Gréville 포인트들이라고 하자. 모든 Gréville 포인트들이 반드시 동시에 선택되지는 않고 (그러나 이것이 또한 가능한데, 그 경우에는 $P = A^{-1}G$ 임), 이는 $V \neq \{0, \dots, n\} \times \{0, \dots, m\}$ 임을 의미하는 것에 주의한다. 이 예의 평활화 원리는, 초기 제어 포인트들 Q_{ij} 에 가능한 가깝게 새로운 제어 포인트들 P_{ij} 를 컴퓨팅하는 것이다.

[0094] 수학적 관점에서, 초기 제어 포인트와 새로운 제어 포인트 사이의 제곱한 거리들의 합산은 이 예에서는 다음과 같이:

[0095]
$$\sum_{i=0}^n \sum_{j=0}^m \|P_{ij} - Q_{ij}\|^2$$

[0096] 새로운 제어 포인트들 P_{ij} 가 변경된 Gréville 포인트들에 순응하는 (선형) 제약 하에서, 즉:

[0097]
$$G_{kl} = \sum_{i=0}^n \sum_{j=0}^m a_{ijkl} P_{ij} \quad (k, l) \in V$$

[0098] 에서 최소화될 수도 있다.

[0099] 이 합산은 추후에 언급되는 가중된 합산으로서 간주될 수 있고, 여기서 모든 가중치들은 1 과 동일하다는 것에 우선 주목한다.

[0100] 또한, 이전의 선형 시스템은 $3(n+1)(m+1)$ 스칼라 미지수들 (제어 포인트 P_{ij} 의 좌표들) 및 $3|V|$ 스칼라 방정식들을 특징으로 하고, 여기서 $|V|$ 는 세트 V 의 엘리먼트들의 수를 표기한다는 것에 주의한다.

모든 Gréville 포인트들이 선택되지 않기 때문에, $|V| < (n+1)(m+1)$ 이다. 그 결과, 이전의 선형 시스템은 방정식들보다 더 많은 미지수들을 특징으로 한다. 이것은 $\tilde{G} = \tilde{A}P$ 로 기재될 수 있고, 여기서 \tilde{A} 는 선형 변환: $\tilde{A}: \mathbb{R}^{3(n+1)(m+1)} \rightarrow \mathbb{R}^{3|V|}$ 이다.

[0101] 수학적 문제는 이제 선형 제약 하에서의 2차 목적 함수의 최소화이다. 다시, Q 는 모든 초기 제어 포인트들의 모든 좌표들의 로우 벡터라고 하자:

[0102]
$$Q = (Q_{ij})_{(i,j) \in \{0, \dots, n\} \times \{0, \dots, m\}} \in \mathbb{R}^{3(n+1)(m+1)}$$

[0103] 표준 수학적 공식은 다음과 같고, 여기서 P 는 미지수이고, Q 및 \tilde{G} 는 입력 벡터들이며 \tilde{A} 는 입력 행렬이다.

$$P = \underset{\tilde{A}P - \tilde{G} = 0}{\text{Argmin}} \frac{1}{2} \|P - Q\|^2$$

[0104] 도 6 은 다음 시퀀스로부터 발생한 NURBS 곡선을 예시한다: (1) 사용자는 Gréville포인트들 G_1 및 G_2 를 선택하고, (2) 사용자는 G_1 를 G'_1 로 이동시킨다. 시스템 관점에서, 선택된 Gréville포인트들 G_2 는 고정 설정되고, 선택되지 않은 Gréville 포인트들 G_1 , G_2 및 G_3 은 이동되어 비-오실레이팅 곡선이 G'_1 및 G_2 에 들어맞다는 것을 보장한다. 도트로 된 폴리라인은 NURBS 곡선의 제어 다각형이다.

[0106] 프로그램은 라그랑주 해결법 (Lagrange solving) 으로 효율적으로 해결될 수도 있다. 이것은 선형-시간 해결법을 허용한다. 실제로, 라그랑주 해결법은 본질적으로 알려진 바와 같이 어픽스 함수 (affix function) 의 파라미터들을 결정하는 것을 포함할 수도 있고, 상기 어픽스 함수는 입력으로서 S40 에서 선택된 표면 포인트들의 서브세트의 위치션을 취급하고 상기 어픽스 함수는 업데이트된 제어 포인트들을 출력한다. 이 방법은 그 후에 어픽스 함수를 선택된 표면 포인트들의 서브세트의 위치션에 수정된 대로 적용하여, 선형-시간에서 결과를 출력할 수도 있다.

[0107] 라그랑주 해결법은 특히 서브-단계들에서의 해결법의 분해를 허용하고, 그 일부의 결과가 재사용되어 효율성을 얻어서, 반응성을 개선시킬 수도 있다. 선택된 표면 포인트들의 서브세트의 위치션을 수정하는 단계 (S52) 및 제어 포인트들의 그리드를 업데이트하는 단계 (S54) 는 실제로 동일한 선택된 표면 포인트의 서브세트를 이용하여 반복될 수도 있다. 다시 말해서, 사용자는 S40 에서 Gréville 포인트들의 세트를 선택하고, 그 후에 이들을 반복된 단계별 동작에서 (예를 들어, 시행착오 방식으로) 이동시킨다. 어픽스 함수의 파라미터들은 이러한 경우 초기 반복시 컴퓨팅 및 저장되고 그 후에 후속 반복들 전반에 걸쳐 추출될 수도 있다. 라그랑주 해결법에 의해 허용된, 그래픽 응답을 최적화시키는 이미 컴퓨팅된 매트릭들의 이 스마트한 재사용은 아래에 상세화된다.

[0108] 심볼릭 레졸루션 (symbolic resolution) 은 라그랑주 멀티플라이어들의 고전적 적용이다. 컴퓨터이션들은 일관성을 위해 재생성된다. 해결 목적만을 위해, 알려지지 않은 라그랑주 멀티플라이어 벡터

$\lambda \in \mathbb{R}^{3|V|}$ 가 도입되어 문제의 라그랑주 함수는 다음과 같다.

$$L(P, \lambda) = \frac{1}{2} \|P - Q\|^2 + \langle \lambda, \tilde{A}P - \tilde{G} \rangle$$

[0109] 문제의 방정식은 다음과 같다:

$$\frac{\partial L}{\partial P}(P, \lambda) = 0$$

$$\frac{\partial L}{\partial \lambda}(P, \lambda) = 0$$

[0111] 행렬 전치를 \cdot^T 로 써서, L 의 편도 함수들은 다음과 같다:

$$\frac{\partial L}{\partial P}(P, \lambda) = P - Q + \tilde{A}^T \lambda$$

$$\frac{\partial L}{\partial \lambda}(P, \lambda) = \tilde{A}P - \tilde{G}$$

[0114] 그 결과, 미지수들 P 및 λ 를 특징으로 하는 선형 시스템은 다음과 같다:

$$P + \tilde{A}^T \lambda = Q$$

$$\tilde{A}P = \tilde{G}$$

[0115]

[0116] 등가 행렬-벡터 공식:

$$\begin{pmatrix} I & \tilde{A}^T \\ \tilde{A} & 0 \end{pmatrix} \begin{pmatrix} P \\ \lambda \end{pmatrix} = \begin{pmatrix} Q \\ \tilde{G} \end{pmatrix}$$

[0117]

[0118] 은 각각의 Gréville 포인트의 단일성 때문에 이 표현의 극좌 행렬이 가역가능하다는 것에 주목한다.

[0119] 이 시스템은 다음과 같이 해결된다. 제 1 방정식을 \tilde{A} 에 의해 곱하면 다음이 산출된다.

$$\tilde{A}P + \tilde{A}\tilde{A}^T \lambda = \tilde{A}Q$$

$$\tilde{A}P = \tilde{G}$$

[0120]

[0121] 그 후에, 제 2 방정식을 사용함으로써 다음과 같다.

$$\tilde{G} + \tilde{A}\tilde{A}^T \lambda = \tilde{A}Q$$

[0122]

[0123] 행렬 $\tilde{A}\tilde{A}^T$ 는 행렬 A 를 가역가능하게 만드는 동일한 이유들로 종래 기술에 따라 가역가능하다. 그 결과, λ 는 다음과 같이 컴퓨팅된다:

$$\lambda = (\tilde{A}\tilde{A}^T)^{-1}(\tilde{A}Q - \tilde{G})$$

[0124]

[0125] 마지막으로, $P = Q - \tilde{A}^T \lambda$ 이기 때문에, 다음과 같다.

$$P = Q - \tilde{A}^T (\tilde{A}\tilde{A}^T)^{-1} (\tilde{A}Q - \tilde{G})$$

[0126]

[0127] 이것은 입력 데이터 Q 및 \tilde{G} 의 역할을 분리시킴으로서 배열될 수 있다:

$$P = (I - \tilde{A}^T (\tilde{A}\tilde{A}^T)^{-1} \tilde{A}) Q + \tilde{A}^T (\tilde{A}\tilde{A}^T)^{-1} \tilde{G}$$

[0128]

[0129] 이러한 방법으로, 상호작용적 애플리케이션을 위해 실시간 피드백을 디스플레이할 때, 첫 번째 항은 일정하게 유지되고 두 번째 항만이 업데이트될 필요가 있다. 일단 NURBS 표면 및 Gréville포인트들이 선택된다면,

제 1 단계는 행렬 $M := \tilde{A}^T (\tilde{A}\tilde{A}^T)^{-1} \tilde{A}$ 및 벡터 $V := (I - M\tilde{A})Q$ 를 컴퓨팅한다. 행렬 M 및 벡터 V 는 Gréville 포인트들 조작 동안 미변경된 채로 유지된다. 그 후에, 사용자가 선택된 Gréville포인트들 \tilde{G} 를 이동시킬 때, 알고리즘은 $P := V + M\tilde{G}$ 를 컴퓨팅함으로써 제어 포인트들 P 를 업데이트한다.

도 7 은 이 예의 데이터 플로우를 예시한다. 정방 행렬 $\tilde{A}\tilde{A}^T$ 의 사이즈는, 일반적으로 매우 작은, 사용자-선택된 Gréville 포인트들의 수에 비례한다는 것에 주목한다.

[0130]

다른 예에서, 프로그램은, 각각의 제어 포인트 각각에 대해, 일반적으로 1 과는 상이할 수도 있는 (즉, 상술된 예에서와는 다름) (예를 들어, 포지티브) 가중치를 이용하여 업데이트 전의 각각의 제어 포인트와 업데이트 후의 각각의 제어 포인트와의 사이의 거리에 제재를 가한다. 이 가중치는 선택된 표면 포인트들의 서브세트와 각각의 제어 포인트에 대응하는 표면 포인트 사이의 거리의 증가 함수의 값에 의해 제공될 수도 있다. 다시 말해서, (예를 들어, 엄격히) 상기 거리가 높을수록, (예를 들어, 엄격히) 가중치가 낮아진다. 상기와

같이, 선택되지 않은 표면 포인트들이 (선택되지 않은 Gréville 포인트들에 대해 이들의 초기 포지션인 사용자 의 요청에 대해) 자유롭게 이동할 수도 있지만, 이제는 Gréville 포인트들이 S40 에서 수행된 선택으로부터 멀어짐에 따라 "감소하는" 자유를 갖는다. 포인트와 포인트들의 세트 사이의 거리는 수학적 방법들로 정의되고 분명해진다. 여기서, 유지된 거리는, 예를 들어, 표면 포인트들의 그리드에 기초한 맨해튼 거리일 수도 있다. 이러한 솔루션은 더욱 직관적인 결과를 발생시킨다. 이 방법은 그래픽 사용자-상호작용에 의해 상기 증가 함수를 파라미터화하는 것을 한층 더 포함할 수도 있다. 따라서, 사용자는 업데이트하는 단계 (S54) 뒤에서 수학적 방법들로 그래픽적으로 상호작용할 수도 있다. 이것은, 슬라이더와 같은, 시스템의 GUI 에 의해 제공된 특정 위젯을 통해 수행될 수도 있다. 이 모두는 "사용자-제어된 미세 튜닝" 을 허용한다.

[0131] 사용자에게 이러한 제어를 제공하기 위해, 소위 "영향 영역", 소위 "영향 비율" 및 소위 "규제 파라미터들" 의 정의들은 이 예에서 이 방법에 의해 사용될 수도 있다. 이 방법의 예에서, 증가 함수는 선택된 Gréville 포인트에 대응하는 각각의 제어 포인트들 각각에 대해 1 과 동일하고, 영향 영역 내측의 Gréville 포인트들에 대응하는 다른 제어 포인트들에 대해 1 보다 더 높은 제 1 의 미리 결정된 수 (사실상 실수) 와 동일하며 (상기 제 1 의 미리 결정된 수는 예들에서 추후에 상세화되는 $1/\alpha$ 임), (영향 영역 외측의 Gréville 포인트들에 대응하는) (모든) 다른 제어 포인트들에 대해 제 2 의 미리 결정된 수 (사실상 실수) 와 동일한 표시 함수이고, 제 2 의 미리 결정된 수는 제 1 의 미리 결정된 수의 적어도 10배보다 더 크지만, 바람직하게는 사실상 무한하다 (즉, 최적화 프로그램에서 이 제 2 의 수에 의해 가중된 제어 포인트들의 임의의 이동을 막는 기능을 갖는 수). 예를 들어, 이 영향 영역은 구체적으로는 표면 포인트들의 그리드에 대해 (즉, 표면 포인트들의 그리드에 기초하여), 선택된 표면 포인트들의 서브세트에 대한 맨해튼 거리가 미리 결정된 임계치 아래에 있는 모든 선택되지 않은 Gréville 포인트들 (및 선택된 Gréville 포인트들) 을 포함할 수도 있다. 그러나, 더욱 정교화되고 사용자 친화적인 다른 예가 추후에 상세화된다. (상술된 슬라이더에 의해) 제 1 의 미리 결정된 수에 대해 그래픽 사용자-상호작용에 의한 파라미터화가 수행될 수도 있다. 사용자에게 대한 부담을 비교적 낮게 유지하기 위해, 컴퓨터 시스템은 미리 결정된 방법으로 모든 다른 파라미터들 (즉, 미리 결정된 임계치 및 제 2 의 수들) 을 핸들링할 수도 있다 (예를 들어, 제 1 의 미리 결정된 수는, 예를 들어, 영향 영역을 정의하는 상기 임계치 파라미터에 관련된다).

[0132] 미세 튜닝이 다른 방법들에 의해 (통상적으로는 가우시안 함수의 합성 곱을 사용함으로써) 구현될 수 있다는 것에 주의해야 한다. 가중치들은 (이미지 프로세싱에서 널리 사용되는 기법인) 가우시안 함수들의 데카르트 곱들로서 정의될 수 있다. 상술된 고유한 임계치 파라미터 대신에, 이러한 경우에는 2개의 파라미터들이 존재한다 (가우시안의 표준 편차 및 진폭). 가중치는 각각의 제어 포인트에 대응하는 Gréville 가로축에서 2D 가우시안의 값에 관련된다. 그러나, 상술되고 아래에 상세화된 기법은 사용자 관점에서 양호한 결과들을 부여한다.

[0133] 다음에서, N_∞ 은 일정한 매우 큰 수이고, 통상적으로는 $N_\infty = 10^{12}$ 그리고 $\varepsilon = \frac{1}{N_\infty}$ 이다. 사용자 관점에서, ε 는 제로의 역할을 한다. 향상된 솔루션으로서, 이 방법은 이 예에서, Gréville 포인트들 선택에 부가적으로, 변형의 범위를 조정하기 위한 사용자-정의된 영향 비율을 제공한다. 슬라이더는, 상술된 임계치들을 진단사적으로 결정할 수도 있는, 영향 비율만을 위해 제공될 수도 있다. 영향 영역은 미리 결정된 방법으로 컴퓨팅될 수도 있다. 이 예의 방법은 그에 따라 도 8 의 플로우차트에 따라 수행할 수도 있다.

[0134] **영향 영역**

[0135] 상술된 바와 같이, 영향 영역은 정교화된 예에서 선택된 비-코너 표면 포인트 상에 센터링된 표면 포인트들의 그리드에서의 가장 큰 정방형에 속하는 모든 표면 포인트들, 또는 선택된 코너 표면 포인트를 포함하는 표면 포인트들의 그리드의 1/4 에 속하는 표면 포인트로 이루어질 수도 있다. 이것은 아래에 상세화된다.

[0136] $T = \{(i, j) \in \{0, \dots, n\} \times \{0, \dots, m\}\}$ 이 Gréville 포인트들 인덱스들의 그리드라고 하고 $(i_0, j_0) \in \{0, \dots, n\} \times \{0, \dots, m\}$ 이 임의적 인덱스라고 하자. 대략 말하면, $Z(i_0, j_0)$ 라고 쓰

여지, (i_0, j_0) 의 영향 존/영역은, T 에 포함될 수 있는 (i_0, j_0) 에서 센터링된 가장 큰 정방형으로서 임의적으로 정의될 수도 있다.

[0137] 더욱 정밀하게는,

[0138] $x \neq 0$ 인 경우 $\delta(x) = 0$ 그리고 $\delta(0) = 1$ 에 의해 정의된 $\delta: \mathbb{R} \rightarrow \{0,1\}$ 이라고 하자.

[0139]
$$k_{min}(i, j) = \min\{\min\{i_0, n - i_0\}, \min\{j_0, m - j_0\}\}$$

[0140] 그리고

[0141]
$$k_{max}(i, j) = \max\{\min\{i_0, n - i_0\}, \min\{j_0, m - j_0\}\}$$

[0142] 라고 하자.

[0143] 이제,

[0144]
$$k_1(i, j) = k_{min}(i, j) + \delta(k_{min}(i, j))k_{max}(i, j)$$

[0145] 라고 하자.

[0146] (i_0, j_0) 가 T 의 코너 인덱스가 아니라는 것을 의미하는, $k_1(i_0, j_0) > 0$ 인 경우, 영향 영역은 다음과 같다:

[0147]
$$Z(i_0, j_0) = \{(i_0 + i, j_0 + j); i, j \in \{-k_1(i_0, j_0), \dots, k_1(i_0, j_0)\} \cap T\}$$

[0148] (i_0, j_0) 가 T 의 코너 인덱스라는 것을 의미하는, $k_1(i_0, j_0) = 0$ 인 경우, 영향 영역은 (i_0, j_0) 를 포함하는 그리드의 1/4 이다:

[0149]
$$Z(i_0, j_0) = \{(i_0 + i, j_0 + j); i, j \in \{-\frac{1}{2} \min\{m, n\}, \dots, \frac{1}{2} \min\{m, n\}\} \cap T\}$$

[0150] 정의에 의해, 인덱스들의 서브세트의 영향 영역 $(k, l) \in V$ 은 모든 영향 영역들의 합집합이다:

[0151]
$$Z(V) = \bigcup_{(k,l) \in V} Z(k, l)$$

[0152] 도 9 는 그리드 (좌측 그리드) 내측의 선택된 Gréville 포인트의 영향 영역 (의 인덱스들) 및 그리드 (중간 및 우측 그리드들) 의 경계 상의 선택된 Gréville 포인트의 영향 영역 (의 인덱스들) 을 예시한다. 선택된 Gréville 포인트들은 "o" 로 마킹되고, 영향 영역은 "x" 및 "o" 포인트들을 포함한다. 도 10 은 2개의 선택된 Gréville 포인트들 (좌측 그리드) 의 영향 영역 (의 인덱스들) 및 선택된 Gréville 포인트들 (우측 그리드) 의 라인의 영향 영역 (의 인덱스들) 을 예시한다. 마지막으로, 사용자 선택된 Gréville 포인트들 $(G_{kl})_{(k,l) \in V}$ 의 인덱스들 V 가 주어지면, 그리드 T 의 인덱스들은 3개의 카테고리들로 배열된다. 영향 영역 $T - Z(V)$ 의 외측, 사용자-선택된 인덱스들 V 및 사용자 선택된 $Z(V) - V$ 가 아닌 영향 인덱스들

[0153] **영향 비율**

[0154] 영향 비율은 사용자-정의된 수 $\alpha \in [\epsilon, 1]$ 이다. 이것은 고유하게 제 1 의 미리 결정된 수를 정의하고, 이 제 1 의 미리 결정된 수는 그에 의해 사용자 설정들을 통해 파라미터화가능하게 된다. 이것은 또한 영향 영역을 간접적으로 정의하는데 사용될 수도 있다. 가장 작은 변형 비율은 $\alpha = \epsilon$ 로 획득된다. 가장

넓은 변형 비율은 $\alpha = 1$ 로 획득된다. 사용자 인터페이스 관점에서, 제로 값이 ε 대신에 디스플레이된다. 이러한 방법으로, 사용자는 $[0,1]$ 간격에서 파라미터를 조작하고, 이는 보다 단순하다.

[0155] **규제 파라미터**

[0156] 다차원 규제 파라미터 μ 가 도입된다. 이것은 이 예에서 사용자에게 숨겨져 있는 내부 모델링 파라미터이다.

[0157]
$$\mu = (\mu_{ij})_{(i,j) \in \{0, \dots, n\} \times \{0, \dots, m\}} \in \mathbb{R}^{(n+1)(m+1)}$$

[0158] 규제 파라미터의 역할은 알려지지 않은 제어 포인트들 P_{ij} 가 어떻게 초기 제어 포인트들 Q_{ij} 에 가깝게 유지되는지를 조정하는 것이다. 앞서 정의된 최소화 프로그램은 다음과 같이 적용된다:

[0159]
$$\sum_{i=0}^n \sum_{j=0}^m \mu_{ij} \|P_{ij} - Q_{ij}\|^2$$

[0160] 미리 결정된 영향 영역 $Z(V)$ 및 사용자 정의된 영향 비율 a 를 사용함으로써 규제 파라미터 μ 의 수치 값이 정의된다.

[0161] • (i, j) 가 영향 영역의 외측에 있다면, 즉, $(i, j) \in T - Z(V)$ 이라면, $\mu_{ij} = N_\infty$ 이다.

[0162] • (i, j) 가 사용자 선택된 Gréville 포인트의 인덱스라면, 즉, $(i, j) \in V$ 라면, $\mu_{ij} = 1$ 이다.

[0163] • (i, j) 가 사용자 선택된 Gréville 포인트의 인덱스를 제외하고 영향 영역 내측에 있다면, 즉, $(i, j) \in Z(V) - V$ 라면, $\mu_{ij} = \frac{1}{\alpha}$ 이다.

[0164] 그 결과, 최소화된 기준들이 기재될 수 있다:

[0165]
$$N_\infty \sum_{(i,j) \in T - Z(V)} \|P_{ij} - Q_{ij}\|^2 + \sum_{(i,j) \in V} \|P_{ij} - Q_{ij}\|^2 + \frac{1}{\alpha} \sum_{(i,j) \in Z(V) - V} \|P_{ij} - Q_{ij}\|^2$$

[0166] 분명하게는, 사용자가 $\alpha = \varepsilon$ 를 설정한다면, 모든 $(i, j) \in T - V$ 에 대해 $\mu_{ij} = N_\infty$ 이고 최소화된 기준들은 다음과 등가이다:

[0167]
$$\sum_{(i,j) \in T - V} \|P_{ij} - Q_{ij}\|^2 + \varepsilon \sum_{(i,j) \in V} \|P_{ij} - Q_{ij}\|^2$$

[0168] 따라서, 영향 영역 외측에 있는 Gréville 포인트들과 연관된 모든 초기 제어 포인트들 Q_{ij} 은 매우 강한 영향을 갖는다. 그래서, 표면 변형은 사용자 선택된 Gréville 포인트들의 이웃으로 좁아진다.

[0169] 한편, 사용자가 $\alpha = 1$ 을 설정한다면, 모든 $(i, j) \in Z(V)$ 에 대해 $\mu_{ij} = 1$ 이고 모든 $(i, j) \in T - Z(V)$ 에 대해 $\mu_{ij} = N_\infty$ 이며 최소화된 기준들은 다음과 등가이다:

[0170]
$$\sum_{(i,j) \in T - Z(V)} \|P_{ij} - Q_{ij}\|^2 + \varepsilon \sum_{(i,j) \in Z(V)} \|P_{ij} - Q_{ij}\|^2$$

[0171] 도 11 내지 도 13 은 일 예시적인 NURBS 곡선에 대한 영향 비율의 형상화 역할을 예시한다. 초기 NURBS 곡

선은 수평 곡선이고 사용자의 액션은 좌측 도면들에 예시된 바와 같이 중간 Gréville 포인트를 수직으로 이동시키는 것이다. 우측 도면은 그의 제어 다각형 (파선들) 을 갖는 결과적인 NURBS 곡선이다. 도 11 에서, 영향 비율은 $\alpha = 1$ 이다. 도 12 에서, 영향 비율은 $\alpha = 0.5$ 이다. 도 13 에서, 영향 비율은 $\alpha = \varepsilon$ 이다. 선택된 Gréville 포인트와 연관된 제어 포인트만이 대체된다는 것에 주의한다. 다른 제어 포인트들은 초기 수평 NURBS 곡선의 것들이다.

[0172] 도 14 내지 도 16 은 일 예시적인 NURBS 표면 변형에 대한 영향 비율의 영향을 예시한다. 초기 NURBS 표면은 평평한 패치이고 사용자의 액션은 선택된 Gréville 포인트들 (볼드 도트들) 을 상향으로 이동시키는 것이다. 명확성을 위해, 제어 포인트들이 예시되어 있지 않다. 도 14 에서, 영향 비율은 $\alpha = 1$ 이다. 도 15 에서, 영향 비율은 $\alpha = 0.5$ 이다. 도 16 에서, 영향 비율은 $\alpha = \varepsilon$ 이다. 형상화 영향이 결과적인 NURBS 표면의 평평해진 코너들에 대해 더욱 가시적이다.

[0173] 위에서 상세화된 바와 같이, 이 예의 프로그램은 다음과 같은 타입의 것이다:

[0174]
$$\sum_{i=0}^n \sum_{j=0}^m \mu_{ij} \|P_{ij} - Q_{ij}\|^2$$

[0175] 여기서 P_{ij} 및 Q_{ij} 는 업데이트 후 그리고 업데이트 전 각각의 제어 포인트들이다. 이러한 2차 목적 함수는 해결하는 것이 간단한데, 이는 이것이 선형 시간에서 해결될 수 있어서 높은 반응성을 제공할 수 있기 때문이다.

[0176] 이 2차 공식이 이하 설명된다.

[0177] 최소화된 기준들의 더욱 적합한 공식은 행렬 $W \in \mathcal{L}(\mathbb{R}^{3(n+1)(m+1)})$ 을 수반하고 그 계수들은 규제 파라미터들 μ_{ij} 이다. 이전에 설명된 바와 같이, 행렬 W 는 규제 파라미터 μ_{ij} 를 통한 영향 비율 a 에 좌우된

다는 것에 주의해야 한다. 이러한 방법으로, 식 $\sum_{i=0}^n \sum_{j=0}^m \mu_{ij} \|P_{ij} - Q_{ij}\|^2$ 은 다음과 등가이다.

[0178]
$$(P - Q)^T W (P - Q)$$

[0179] 이전의 정의로부터, W 가 대각인 것이 분명하다.

[0180] 예를 들어, $P_{ij} - Q_{ij} = X_{ij} = \begin{pmatrix} x_{ij} \\ y_{ij} \\ z_{ij} \end{pmatrix} \in \mathbb{R}^3$ 인 것으로 쓰고 $n = m = 1$ 을 선정하여, 오리지널 식은 다음과 같다:

[0181]
$$\begin{aligned} & \sum_{i=0}^1 \sum_{j=0}^1 \mu_{ij} \|P_{ij} - Q_{ij}\|^2 \\ &= \mu_{00}(x_{00}^2 + y_{00}^2 + z_{00}^2) + \mu_{01}(x_{01}^2 + y_{01}^2 + z_{01}^2) \\ &+ \mu_{10}(x_{10}^2 + y_{10}^2 + z_{10}^2) + \mu_{11}(x_{11}^2 + y_{11}^2 + z_{11}^2) \end{aligned}$$

[0182] 그리고 행렬 $W \in \mathcal{L}(\mathbb{R}^{12})$ 를 특징으로 하는 등가의 단순한 식은 다음과 같다:

$$\begin{pmatrix} x_{00} \\ y_{00} \\ z_{00} \\ x_{01} \\ y_{01} \\ z_{01} \\ x_{10} \\ y_{10} \\ z_{10} \\ x_{11} \\ y_{11} \\ z_{11} \end{pmatrix}^T \begin{pmatrix} \mu_{00} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \mu_{00} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \mu_{00} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu_{01} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \mu_{01} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu_{01} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \mu_{10} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mu_{10} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mu_{10} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mu_{11} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mu_{11} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mu_{11} \end{pmatrix} \begin{pmatrix} x_{00} \\ y_{00} \\ z_{00} \\ x_{01} \\ y_{01} \\ z_{01} \\ x_{10} \\ y_{10} \\ z_{10} \\ x_{11} \\ y_{11} \\ z_{11} \end{pmatrix}$$

[0183]

[0184] 행렬 W 가 대각 행렬이므로, $W = W^T$ 이다. 또한, 행렬 W 는 규제 파라미터들이 영이 되지 않기 때문에 가역가능하다.

[0185] 이 예의 라그랑주 해결법이 이제 설명되는데, 이는 라그랑주 해결법이 앞선 비-가중된 예에 대해 설명되었기 때문이다 (즉, 가중치들 μ_{ij} 모두가 1 과 동일함).

[0186] 여기에서 다시, 심볼릭 레졸루션은 라그랑주 멀티플라이어들의 고전적 적용이다. 컴퓨터이션들은 일관성을 위해 재생성된다. 해결 목적만을 위해, 알려지지 않은 라그랑주 멀티플라이어 벡터 $\lambda \in \mathbb{R}^{3|V|}$ 가 도입되어 문제의 라그랑주 함수는 다음과 같다:

$$L(P, \lambda) = \frac{1}{2} (P - Q)^T W (P - Q) + \langle \lambda, \tilde{A}P - \tilde{G} \rangle$$

[0187]

[0188] 문제의 방정식은 다음과 같다:

$$\begin{aligned}
 \frac{\partial L}{\partial P}(P, \lambda) &= 0 \\
 \frac{\partial L}{\partial \lambda}(P, \lambda) &= 0
 \end{aligned}$$

[0189]

[0190] L 의 편도 함수들은 다음과 같다:

$$\begin{aligned}
 \frac{\partial L}{\partial P}(P, \lambda) &= W(P - Q) + \tilde{A}^T \lambda \\
 \frac{\partial L}{\partial \lambda}(P, \lambda) &= \tilde{A}P - \tilde{G}
 \end{aligned}$$

[0191]

[0192] 첫 번째 라인은 $W(P - Q + W^{-1}\tilde{A}^T\lambda)$ 로 기재될 수 있어서 $W(P - Q) + \tilde{A}^T\lambda = 0$ 이 $P - Q + W^{-1}\tilde{A}^T\lambda = 0$ 과 동가이다. 그 결과, 미지수들 P 및 λ 를 특징으로 하는 선형 시스템은 다음과 같다:

$$\begin{aligned}
 P + W^{-1}\tilde{A}^T\lambda &= Q \\
 \tilde{A}P &= \tilde{G}
 \end{aligned}$$

[0193]

[0194] 등가 행렬-벡터 공식:

$$[0195] \begin{pmatrix} I & W^{-1}\tilde{A}^T \\ \tilde{A} & 0 \end{pmatrix} \begin{pmatrix} P \\ \lambda \end{pmatrix} = \begin{pmatrix} Q \\ \tilde{G} \end{pmatrix}$$

[0196] 이 시스템은 다음과 같이 해결된다. 제 1 방정식을 \tilde{A} 에 의해 곱하면 다음이 산출된다:

$$[0197] \begin{aligned} \tilde{A}P + \tilde{A}W^{-1}\tilde{A}^T\lambda &= \tilde{A}Q \\ \tilde{A}P &= \tilde{G} \end{aligned}$$

[0198] 그 후에, 제 2 방정식을 사용함으로써 다음과 같다:

$$[0199] \tilde{G} + \tilde{A}W^{-1}\tilde{A}^T\lambda = \tilde{A}Q$$

[0200] 행렬 $\tilde{A}W^{-1}\tilde{A}^T$ 는 행렬 $\tilde{A}\tilde{A}^T$ 가 가역가능하기 때문에 가역가능하여, λ 가 다음과 같이 컴퓨팅된다:

$$[0201] \lambda = (\tilde{A}W^{-1}\tilde{A}^T)^{-1}(\tilde{A}Q - \tilde{G})$$

[0202] 마지막으로, $P = Q - W^{-1}\tilde{A}^T\lambda$ 이기 때문에, 다음과 같다.

$$[0203] P = Q - W^{-1}\tilde{A}^T(\tilde{A}W^{-1}\tilde{A}^T)^{-1}(\tilde{A}Q - \tilde{G})$$

[0204] 이것은 입력 데이터 Q 및 \tilde{G} 의 역할을 분리시킴으로서 배열될 수 있다:

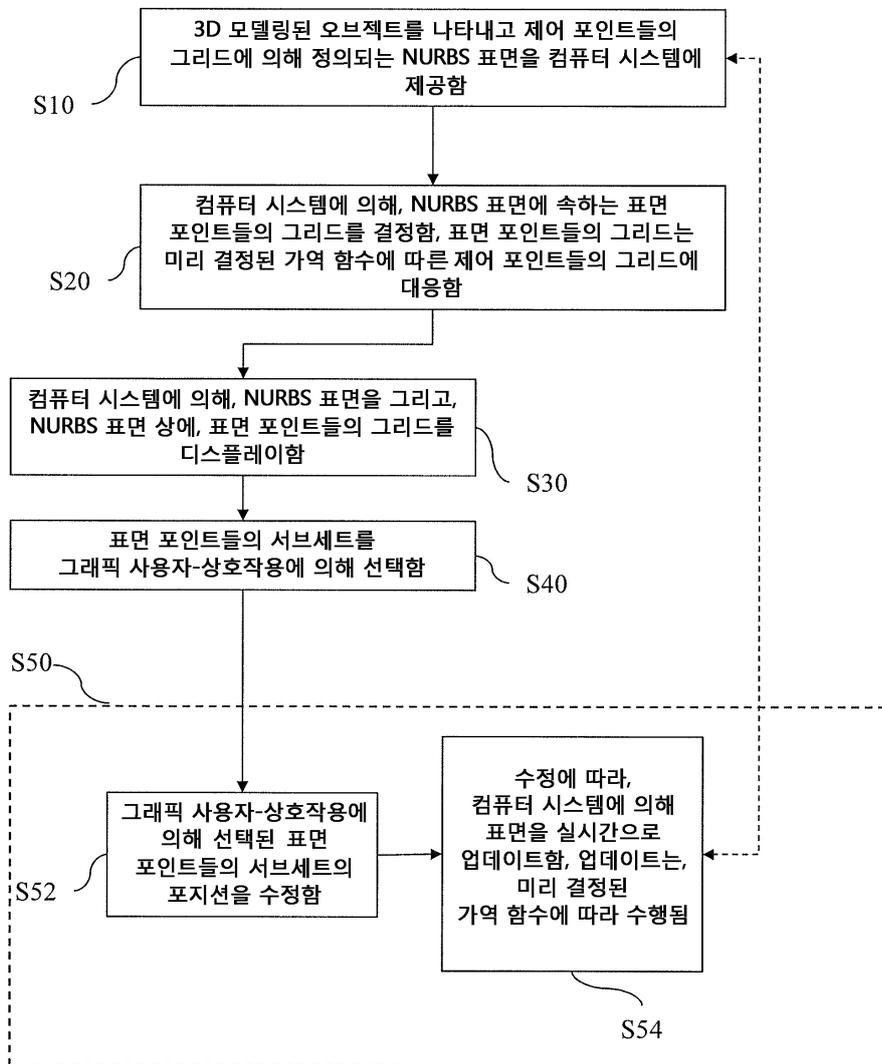
$$[0205] P = (I - W^{-1}\tilde{A}^T(\tilde{A}W^{-1}\tilde{A}^T)^{-1}\tilde{A})Q + W^{-1}\tilde{A}^T(\tilde{A}W^{-1}\tilde{A}^T)^{-1}\tilde{G}$$

[0206] 이러한 방법으로, 실시간 피드백을 디스플레이할 때, 첫 번째 항은 Gréville 포인트들이 이동될 때에 일정하게 유지된다. 역으로, 영향 비율 a 가 변경될 때, 규제 파라미터들 μ_{ij} , 그리고 그에 따라 행렬 $\#$ 가 a 에 좌우되기 때문에 양쪽 항들이 업데이트된다.

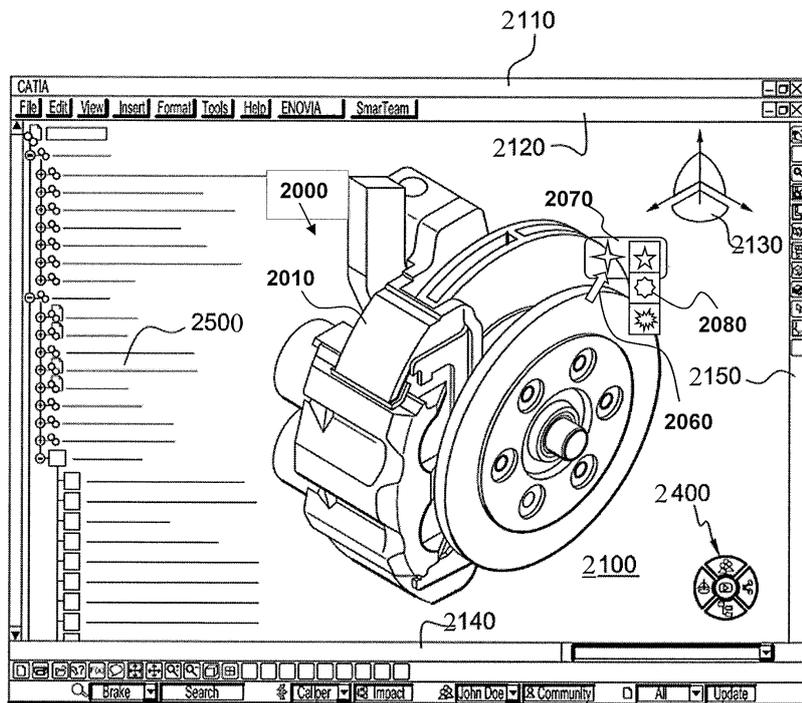
[0207] 일단 NURBS 표면 및 일부 Gréville 포인트들이 선택되고, 영향 비율 a 에 대한 디폴트 값이 설정된다면, 제 1 단계는 행렬 $M := W^{-1}\tilde{A}^T(\tilde{A}W^{-1}\tilde{A}^T)^{-1}$ 및 벡터 $V := (I - M\tilde{A})Q$ 를 컴퓨팅한다. 행렬 M 및 벡터 V 는 Gréville 포인트들 조작 동안 미변경된 채로 유지된다. 그 후에, 사용자가 선택된 Gréville 포인트들 \tilde{G} 를 이동시킬 때, 알고리즘은 $P := V + M\tilde{G}$ 를 컴퓨팅함으로써 제어 포인트들 P 를 업데이트 한다. 사용자가 영향 비율 a 를 변경할 때, 행렬 M 이 a 에 좌우되기 때문에 모든 항들이 업데이트를 필요로 한다. 도 17 은 데이터 플로우를 예시한다.

도면

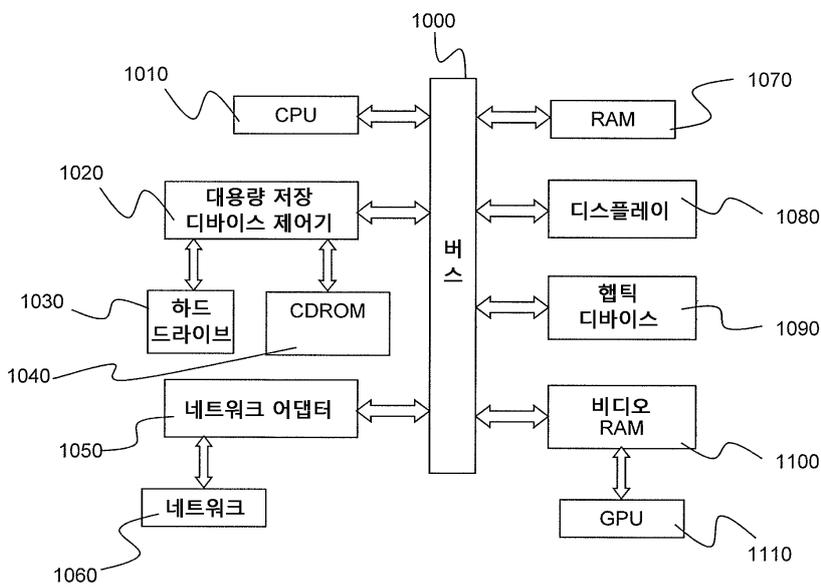
도면1



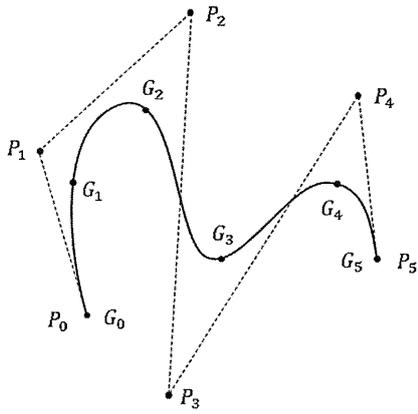
도면2



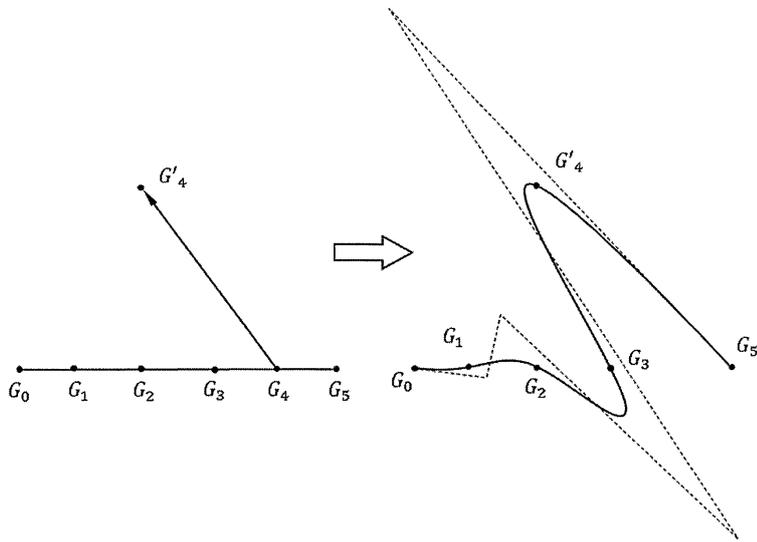
도면3



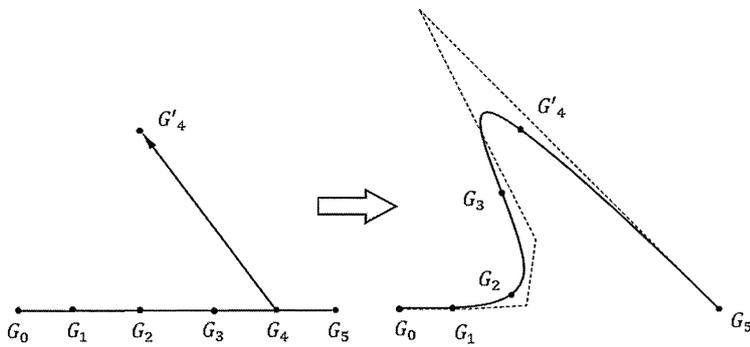
도면4



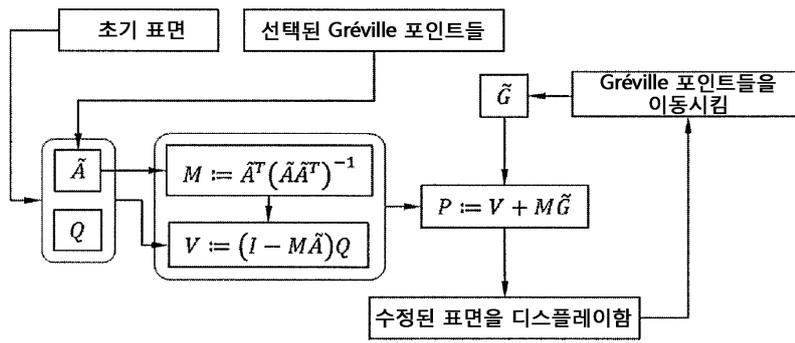
도면5



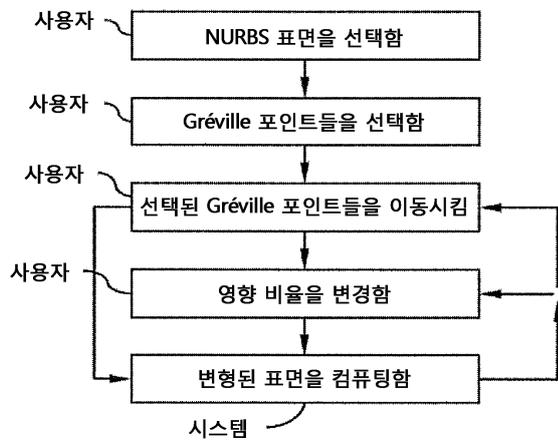
도면6



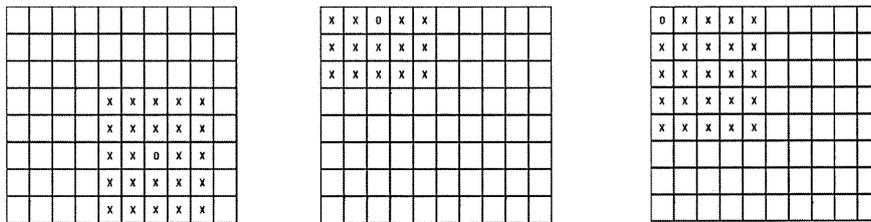
도면7



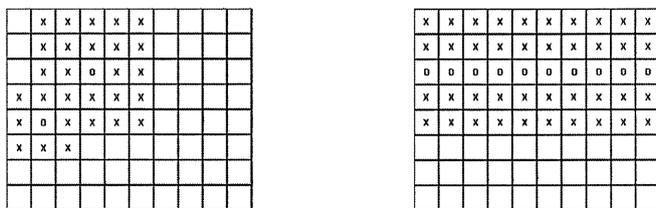
도면8



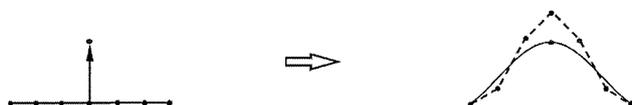
도면9



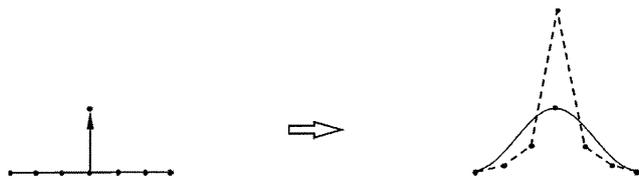
도면10



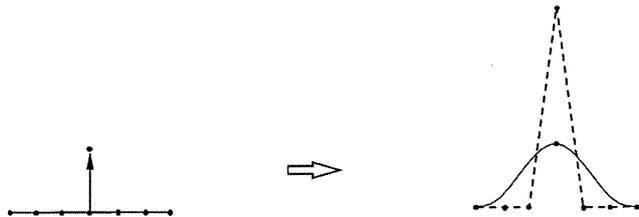
도면11



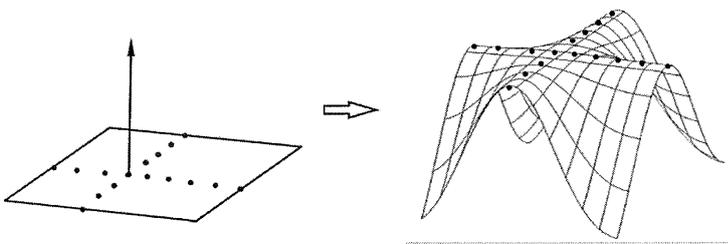
도면12



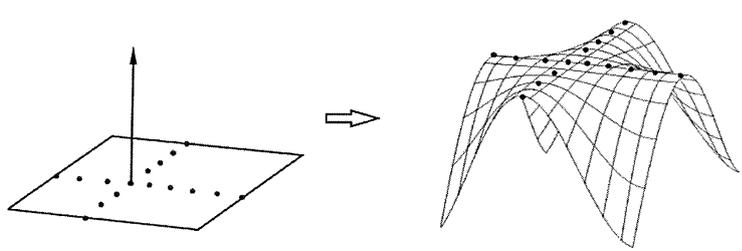
도면13



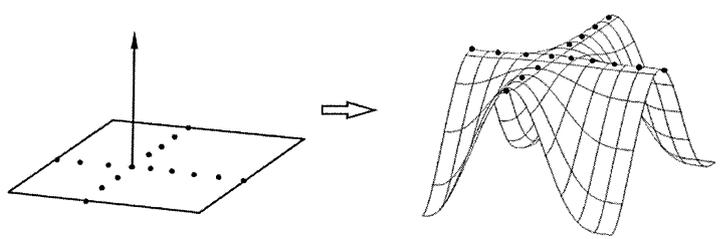
도면14



도면15



도면16



도면17

