

다상 기체의 형상 제어를 위한 파티클 기반 유체 애니메이션

노병석^o, 김창헌
고려대학교 그래픽스 연구실
{prayforher, chkim}@korea.ac.kr

Particle based fluid animation for controllable multiphase smoke

Byungseok Roh^o, Changhun Kim
Korea University Graphics Laboratory

요 약

본 논문은 파티클을 이용하여 다상의 기체에 대한 형상 제어를 수행하는 기법을 제안한다. 유체를 하나의 목표 형상으로 변형시키기 위한 힘만 셀에 적용하는 기존 연구와 달리, 본 논문에서는 단일 밀도장내에서 파티클마다 다른 목표 형상으로 가고자 하는 힘을 정의하여 서로 다른 성질의 유체를 구성하는 파티클이 각자의 목표 형상으로 진행되는 동시에 상호작용이 가능하도록 하였다. 또한, 목표 형상 내부에서의 분산력을 정의하여 파티클들이 목표 형상에 도달한 후에도 목표 형상의 세밀한 특징을 표현하기 위해 멈추지 않고 고르게 분산되게 하였다.

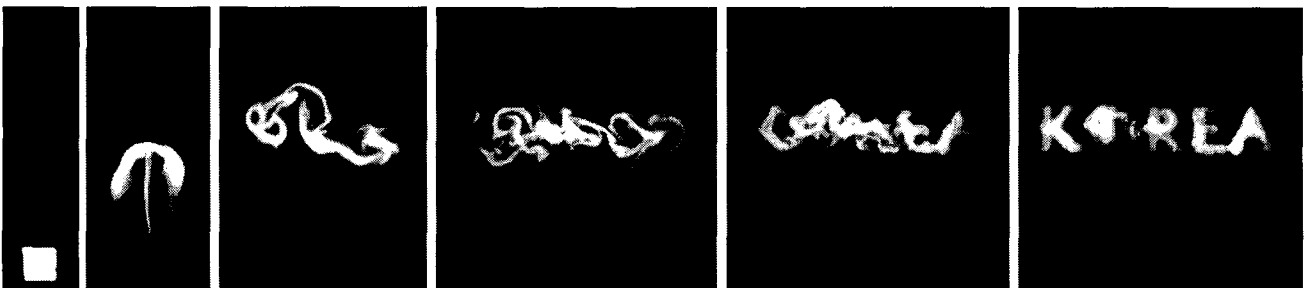


그림 1. 초기 상태에서부터 서로 다른 목표 형상을 찾아 가는 모습

1 Introduction

지난 몇 년간 액체나 기체와 같은 자연 현상을 사실적으로 시뮬레이션하고 가시화하는 유체 애니메이션에 관한 많은 연구들이 진행되어 왔다. 최근 들어서는 사실적인 시뮬레이션에 그치지 않고, 애니메이터

의 의도에 따라 유체의 형상 제어를 수행하는 연구들이 새롭게 각광받고 있다.

그동안 3차원 애니메이션이나 영화와 같은 분야에서 유체를 가시화하고 형상제어를 수행하는 기법은 많이 시도되었지만, 대개의 경우 물리적인 법칙을 무시한 채로 애니메이터의 키프레임 정의에 의한 수작

업으로 생성되는 경우가 많았다. 그러나 최근 물리법칙에 따라 유체의 사실적인 움직임을 생성하여 목표 형상으로 변화시키는 연구들이 발표됨으로써 유체 역학에 근거한 사실적인 형상제어 시뮬레이션이 가능하게 되었다.

이러한 유체의 형상 제어에 관련된 기존 연구에서는 애니메이터가 초기 상태와 목표 형상만 제시하면 유체가 물리적인 법칙에 따라 사실적인 움직임을 보이며 목표 형상에 수렴하도록 하였다. 그러나 목표 형상까지의 거리 장을 구성한 후, 가장 가까운 형상으로 찾아가도록 힘을 가해 유체를 제어하기 때문에 애니메이터가 원치 않는 결과를 가져오거나 그 형상의 변화에 있어서 한계가 있었다. 기존의 연구는 각각의 파티클이 현재 위치에서 가까운 목표 형상만을 향해가기 때문에 그림 2에서 하단의 두 기체가 서로 교차되면서 대각선 방향으로 이동하여 목표 형상에 수렴하길 원한다 하더라도 가장 가까운 목표 형상을 향해 수직으로 상승하는 움직임밖에 보여줄 수 없다는 한계가 있었다.

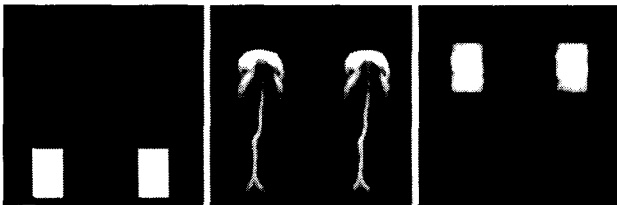


그림 2. 유체 형상 제어의 기존 연구 결과

우리의 방법은 기본적으로 [Hong04]의 연구와 비슷하게 목표 형상에 이르는 거리를 사용하여 기체를 제어하였다. 그러나 기존의 연구는 하나의 형상을 제어하는 방법만 제공해 주었지만, 우리는 여러 개의 목표 형상을 제어할 수 있는 새로운 방법을 소개할 것이다.

2 Related Work

지난 몇 년간 전산 유체 역학(CFD)에 기반하여 사실적인 유체의 움직임을 생성하는 다양한 방식의 연구가 진행되어 왔다. [Stam99, Fedkiw01]

[Foster97]는 애니메이터로 하여금 유체의 속성을 제어하는 매개변수들을 조절하여 다양한 성질의 유체 움직임을 생성하였다. 그러나 이 연구에서는 애니메이터들이 원하는 움직임의 유체를 생성하기 위해서 그에 관련된 역학을 이해하고 있어야만 했고, 형상 자체를 제어하지는 못했다.

[Treuille03]은 원하는 목표 형상을 키프레임으로 입력받아 각각의 키프레임으로 변화하기 위한 최적화된 바람의 방향 및 힘을 계산하여, 기체의 형상 제어를 수행하였다. 그러나 이 논문에서는 최적화된 바람의 방향 및 힘을 구하기 위한 목적 함수(Objective Function)를 계산하기 위해서 상당히 많은 시간이 소요되는 단점이 있었다.

[Fattal04, Hong04]는 목표 형상에 대한 거리 장을 바탕으로 유체를 제어하는 방법을 제안하였다. 이러한 방법은 계산 복잡도가 크지 않고, 애니메이터가 유체 역학을 전혀 모르는 상황에서 쉽게 사용할 수 있는 장점이 있다. 그러나 이러한 방법은 유체를 목표 형상으로 수렴시키는 외부 힘을 오직 거리 장에만 기초하여 계산하기 때문에, 애니메이터의 의도를 정확히 표현할 수는 없었다.

본 논문에서는 애니메이터의 의도를 직관적으로 입력받아, 기체가 목표 형상으로 수렴하도록 하는 사실적인 움직임을 빠르고 정확하게 표현할 수 있도록 하는 방법을 제안한다. 또한, 기존의 연구에서 볼 수 없었던 여러 개의 목표 형상으로 다상의 기체가 수렴하도록 제어하는 방법을 소개한다.

3 Overview

사실적인 기체의 움직임을 시뮬레이션하기 위해서는 Navier-stokes 또는 Euler 방정식을 수치적으로 근사하여 계산하여야 한다. 우리는 다상의 기체를 시뮬레이션하기 위해 비점성 유체를 다루는 Euler 방정식을 사용하였다. 시간 t 에서 유체의 속도 장을 $u = u(x, t)$ 라 하고 이것을 시간에 대해서 미분한 것을 u_t 라 할 때, 비압축 조건을 만족하는 Euler 방정식은 다음과 같다.

$$u_t = -u \cdot \nabla u - \nabla p + f \quad (1)$$

$$\nabla \cdot u = 0 \quad (2)$$

이때 $p = p(x, t)$ 는 유체 정역학적인 압력이고, f 는 중력, 부력과 같이 유체의 흐름에 영향을 주는 외부 힘을 의미한다. 식 (1)은 유체의 흐름을 정의하는 Euler 방정식이며, 식 (2)는 유체의 부피 보존에 관한 식이다.

식 (1)에서 외부 힘 f 는 유체를 제어할 때 중요하게 쓰이는 변수이며, [Fattal04, Hong04]등에서 f 를 적절히 조절하여 기체를 목표 형상으로 제어하였다. 그러나 앞에서 언급한 바와 같이 기존 연구에서는 단일 목표 형상으로의 제어를 다루었을 뿐, 다상의 목표 형상에 대해서 다루지는 않았다.

본 논문에서는 다상의 목표 형상을 제어하기 위해 가한 힘 f 에 대해서 자세하게 설명할 것이며, 목표 형상을 좀 더 정확하게 찾아가기 위한 새로운 방법인 *Internal target force*에 대해서도 설명할 것이다.

3.1 Target forces on cell

이 장에서는 기체를 목표 형상으로 제어하기 위한 힘(F_T)에 대해서 설명할 것이다.

힘 F_T 를 계산하기 위해서는 우선 목표 형상 모델에 대해서 다음과 같이 거리 장(distance field) D_i ($1 \leq i \leq$ 목표 형상의 개수 P_n)를 생성시켜 주어야 한다.

거리 장은 각각의 셀에서 목표 형상에 이르는 거리 값이 들어가며, 그림 3은 2차원 거리 장의 예를 보여주고 있다. 그림 3에서 볼 수 있듯이 목표 형상은 0으로 표기되며, 이러한 거리 장은 목표 형상의 개수만큼 구해져야 한다.

이렇게 거리 장을 구하게 되면 목표 형상으로 제어하는 힘 F_{T_i} 는 식 (3)에서와 같이 거리 장 D_i 의 gradient 값을 계산함으로써 구할 수 있다.

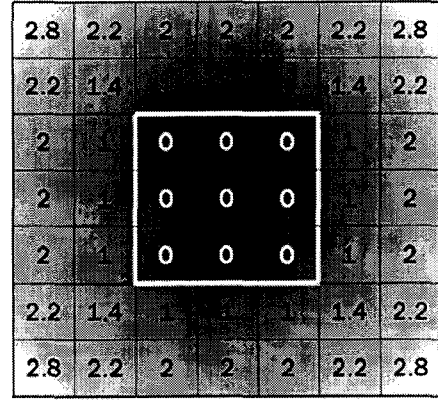


그림 3. 2차원 distance field의 예 (내부의 흰색 사각형 : 목표 형상)

$$F_{T_i} = -\nabla D_i \quad (3)$$

F_{T_i} 는 거리가 감소하는 방향(목표 형상으로의 방향)으로의 힘을 나타내야 하기 때문에 증가하는 방향의 값을 가지는 ∇D_i 에 -1을 곱해주어야 하며, F_{T_i} 값 역시 D_i 와 마찬가지로 목표 형상의 개수만큼 구해져야 한다.

기체를 i 번째 목표 형상으로 제어하는 힘 F_{T_i} 를 계산한 다음에는 셀 x 에 들어있는 파티클이 어떠한 방향으로 움직일지 정하는 $F_T(x)$ 를 계산하여야 한다.

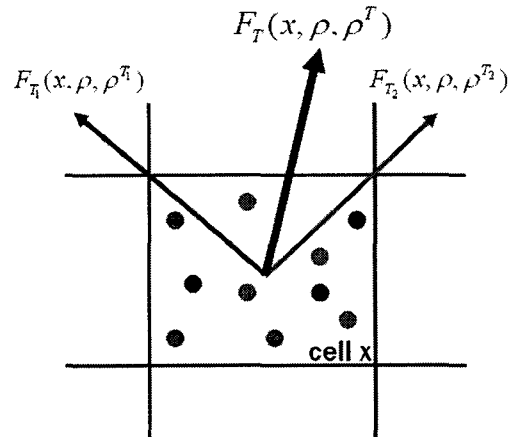


그림 4. $F_T(x, \rho, \rho^T)$ 를 구하는 과정

시간 t 에서의 기체 밀도를 $\rho(x, t)$ 라 하고 우리가

최종적으로 도달해야 하는 목표 형상의 기체 밀도를 $\rho^T(x)$ 라 할 때, 시간 t 에서 셀 x 에 주어져야 하는 힘 $F_T(x)$ 는 $F_T(x, \rho, \rho^T)$ 로 쓰여 질 수 있으며 다음과 같이 정의된다.

$$F_T(x, \rho, \rho^T) = \sum_{i=1}^{P_n} \left(\frac{\sum_{p \in x} p_i}{\sum_{p \in x} p} \times F_{T_i}(x, \rho, \rho^{T_i}) \right) \quad (4)$$

이 때 $\sum_{p \in x} p$ 은 셀 x 에 존재하는 모든 파티클의 개수이며, $\sum_{p \in x} p_i$ 는 셀 x 에 존재하는 i 번째 파티클의 개수이다.

그림 4에서 파란 파티클은 F_{T_1} 방향에 목표 형상이 존재하고 빨간 파티클은 F_{T_2} 방향에 목표 형상이 존재할 때, 셀 x 에 파란 파티클이 3개 있고 빨간 파티클이 7개가 있으므로 셀 x 에 주어지는 힘은

$$F_T(x, \rho, \rho^T) = 0.3F_{T_1}(x, \rho, \rho^{T_1}) + 0.7F_{T_2}(x, \rho, \rho^{T_2})$$

가 된다.

3.2 Particle target force

여러 개의 목표 형상을 제어하기 위해서 셀 x 에 힘 $F(x, \rho, \rho^T)$ 만을 가해주게 되면, 적은 양의 파티클은

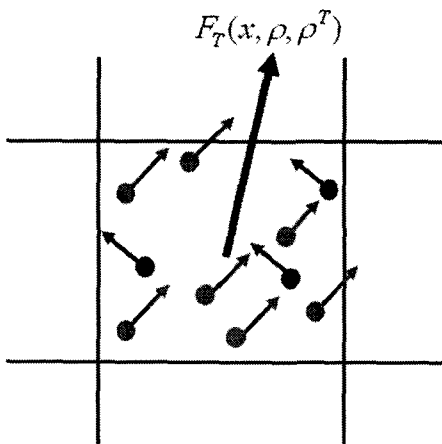


그림 5. 각각의 파티클이 target force를 가지고 움직이는 모습

자신의 목표 형상으로 움직이지 못하고 목표 형상으로부터 점점 더 멀어지는 경우가 생길 수 있는데 이러한 현상을 방지하기 위해서 *particle target force* 라는 힘을 가한다.

*Particle target force*는 그림 5에서와 같이 셀에 가해지는 힘 $F_T(x, \rho, \rho^T)$ 와 비교하여 매우 작은 힘이며, 셀에 가해지는 힘에 저항하여 다른 셀로 이동할 수 있는 가능성을 제공해 준다.

3.3 Internal Target Force

기존의 유체 제어 논문 [Fattal04, Hong04]에서는 목표 형상으로 움직이는 모습만 제공했을 뿐, 목표 형상 내부에서의 움직임을 정의하지 않았기 때문에 복잡한 목표 형상을 표현하는데 한계점이 있었으나 우리의 방법에서는 목표 형상 내부에서의 움직임을 정의하여 복잡하고 날카로운 목표 형상까지도 표현할 수 있다.

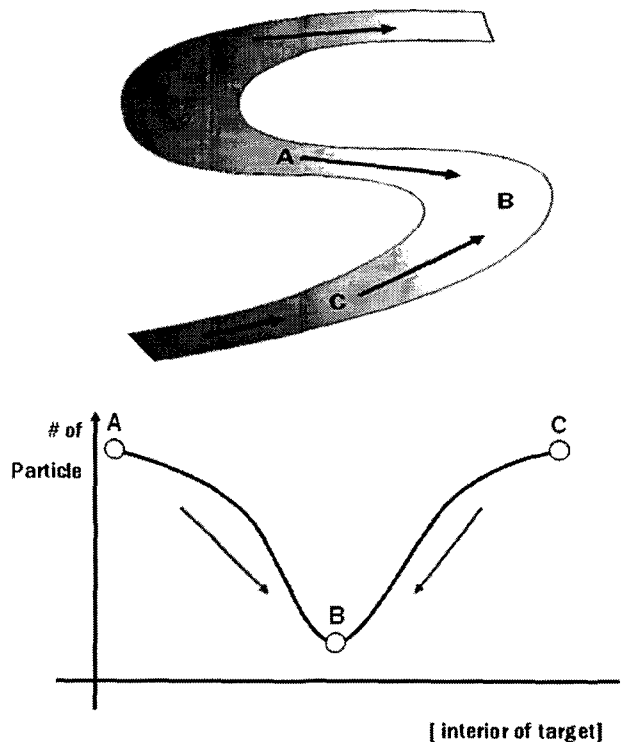


그림 6. 목표 형상 내부에서 주어지는 힘 (색이 붉을수록 고밀도의 영역)

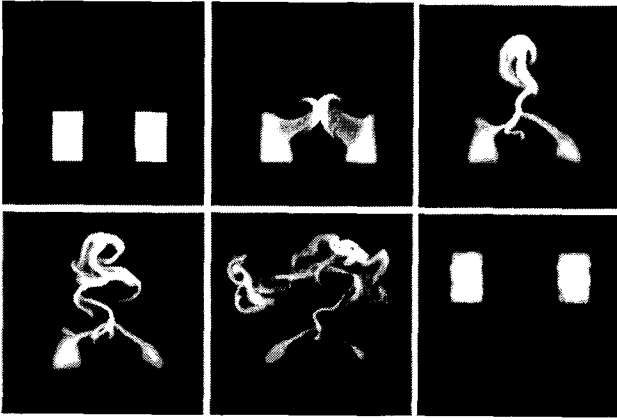


그림 7. 원하는 방향으로 기체를 제어하는 모습

시간 t 에서의 기체 밀도를 $\rho(x,t)$ 라 할 때 Internal target force F_I 는 다음과 같이 정의된다.

$$\begin{cases} F_I = -\nabla\rho & \text{목표 형상 내부} \\ F_I = 0 & \text{목표 형상 외부} \end{cases} \quad (5)$$

F_I 는 목표 형상 내부에서 파티클이 감소하는 방향으로 힘을 가해줌으로써 목표 형상 내부에서 파티클이 많은 곳에서 파티클이 적은 곳으로 파티클을 이동시켜 파티클이 목표 형상 전체에 고르게 분포될 수 있도록 한다.

그림 6에서 보면, A와 C는 파티클이 많이 분포하고 B에는 파티클이 적게 분포하기 때문에, A와 C에서 B로 힘 F_I 가 작용한다.

3.4 Modified equations of Flow

우리의 방법은 식 (1)을 다음과 같이 변형시킨다.

$$u_t = -u \cdot \nabla u - \nabla p + \alpha F_T(\rho, \rho^T) + \beta F_I(\rho, \rho^T) + F_I \quad (6)$$

식 (6)은 기존의 Euler 방정식에 3가지의 외부 힘 f 를 추가시킨 것이다. 3.2장에서 언급했듯이 β 는 α 에 비해서 매우 작은 값이며, 힘 F_I 를 통하여 파티클이

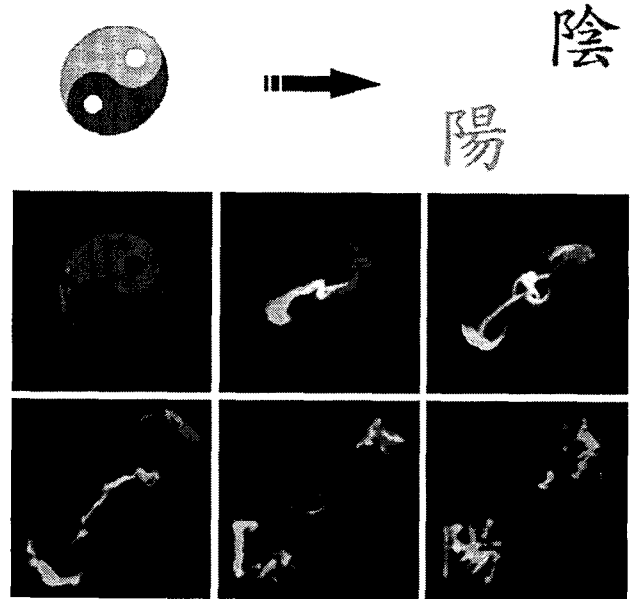


그림 8. 파티클이 서로 상호작용 하는 모습

목표 형상 내부에서 고르게 퍼질 수 있도록 해준다.

4 Result and Discussion

본 연구는 MS Visual C++ 7.0을 사용하여 구현하였으며, 3차원 렌더링의 경우 Maya 6.5를 사용하여 가시화하였다. 모든 결과는 Windows PC Pentium IV 3.2GHz, 1GB 메모리의 컴퓨터에서 수행되었다. 2차원 시뮬레이션을 위해 300^2 격자를 사용하였으며 프레임 당 생성시간은 0.5초가 소요되었다. 3차원 시뮬레이션을 위해서는 100^3 격자가 사용되었으며 프레임 당 생성시간은 11초 가량 소요되었다. 이는 아무런 외력의 작용없이(F_T 및 F_I 제외) 시뮬레이션을 수행한 결과보다 약 20%정도 밖에 느려지지 않았으며, 우리의 방법을 기존의 시스템에 덧붙여 사용하



그림 9. 서로 다른 목표 형상을 찾아가는 과정

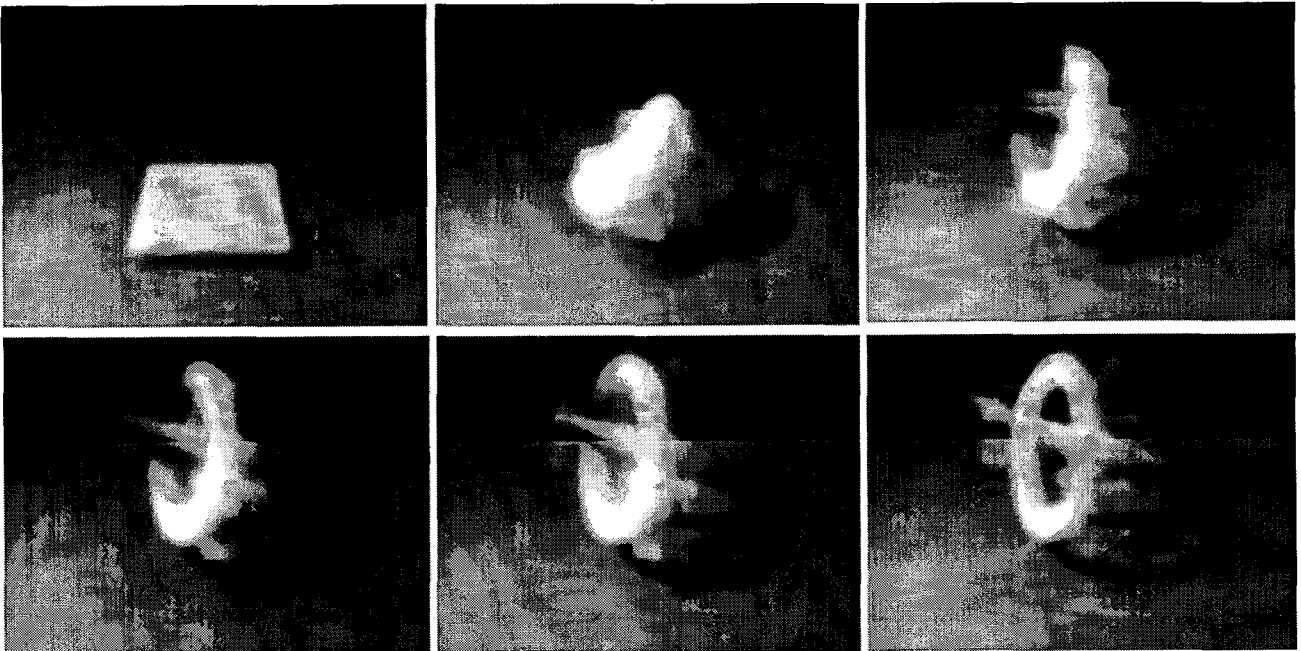


그림 10. 3차원 환경에서 목표 형상을 찾아가는 과정

라도 적은 비용으로 이러한 효과를 낼 수 있음을 보여주고 있다.

그림 7은, 기존 연구(그림 2)에서 표현할 수 없었던 하단의 두 기체가 서로 교차되면서 상단의 기체로 이동하는 과정을 보여주고 있다. 이 결과에서 볼 수 있듯이 우리의 방법을 사용하게 되면 기존 연구에서 표현하지 못했던 애니메이터의 의도를 쉽게 표현할 수 있다.

그림 1, 그림 8, 그리고 그림 9는 2차원 환경에서 목표 형상을 찾아가는 과정을 보여주고 있다. 특히 그림 8는 단일 밀도 장을 사용하였기 때문에 각각의 파티클들이 서로 상호 작용하여, 태극 문양의 가운데 부분에서 노란색 소용돌이무늬가 생김을 볼 수 있다.

마지막으로 그림 10은, 3차원 환경에서 비행기 모델이 링 모양의 모델을 통과하는 형상으로 찾아가는 과정을 보여주고 있다.

5 Conclusion

본 논문은, 기존의 연구에서 가장 가까운 목표 형상을 향하는 움직임밖에 보여줄 수 없었던 단점을 극

복하여 애니메이터의 의도를 충분히 표현할 수 있는 방법을 제안하였다. 또한, 기존의 연구에서 볼 수 없었던 여러 개의 목표 형상으로 다상의 기체가 수렴하도록 제어하는 방법을 제안하였다.

게다가 우리의 방법은 기존에 존재하는 유체 시뮬레이션 기법에 외부 힘 f 만 추가하면 되기 때문에 확장이 매우 용이하다.

향후 연구과제로는 서로 다른 매질을 가진 다상의 액체에 대한 형상 제어 애니메이션을 생성하고자 한다.

References

- [1] Enright, D., Marschner, S., Fedkiw, R. 2002. Animation and rendering of complex water surfaces. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2002)* 21, 3 (July), 736-744.
- [2] Fattal R., Lischinski D. 2004. Target-

- driven smoke animation. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2004)* 23, 3 (July), 439-446.
- [3] Fedkiw, R., Stam, J., Jensen, H. W. 2001. Visual simulation of smoke. In *Computer Graphics Proceedings, Annual Conference Series*, E. Fiume, Ed., ACM SIGGRAPH, 15-22.
- [4] Foster, N., Metaxas, D. 1997. Controlling fluid animation. In *Proceedings of Computer Graphics International '97*, 178-188.
- [5] Foster, N., Fedkiw, R. 2001. Practical animations of liquids. In *Computer Graphics Proceedings, Annual Conference Series*, E. Fiume, Ed., ACM SIGGRAPH, 23-30.
- [6] Hong J.-M., Kim C.-H. 2004. Controlling fluid animation with geometric potential. *Computer Animation and Virtual Worlds* 15, 3-4(2004), 147-157.
- [7] Hong J.-M., Kim C.-H. 2005. Discontinuous Fluids *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2005)* 24, 3 (July), 915-920.
- [8] Lin Shi, Yizhou Yu, Proceedings of the 2005. Taming Liquids for Rapidly Changing Targets. *ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 229-236.
- [9] Treuille, A., Mcnamara, A., Popovic, Z., Stam, J. 2003. Keyframe control of smoke simulations. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2003)* 22, 3 (July), 716-723.
- [10] Stam, J. 1999. Stable fluids. In *Computer Graphics Proceedings, Annual Conference Series*, A. Rockwood, Ed., ACM SIGGRAPH, 121-128.