표시입자가 적용된 FDS 코드 개발

최 춘 범,¹ 장 용 준,² 이 경 준,¹ 양 경 수^{*3}

FIRE DYNAMIC SIMULATOR WITH MARKER PARTICLES

C.-B. Choi,¹ Y.-J. Jang,² K. Lee¹ and K.-S. Yang^{*3}

In this study, Fire Dynamic Simulator (FDS) has been improved with the implementation of marker particles. Convectional FDS is able to implement only two curved geometries, namely, circular cylinders and spheres. The introduction of marker particles made it possible to implement arbitrarily curved objects. Moreover, an algorithm that extracts the marker particle information from a model designed by AutoCAD was developed. With the present implementation, applicability of FDS is greatly enhanced, beneficial to both academia and industry.

Key Words : FDS(Fire Dynamic Simulator), 표시입자(marker particles), 컴퓨터 지원설계 프로그램(AutoCAD), 임의 곡면 형상(arbitrarily curved objects)

1. 서 론

2003년 대구 지하철 참사를 통해 폐쇄된 지하공간에서 방화 등으로 인한 유독가스의 위험성이 대두된 바 있다. 도시철도 노선의 증설로 인해 지하역사는 더욱 깊어지고 있으며 피난에 필요한 거리와 시간 역시 증대되고 있다. 따라서 폐쇄 공간 내에서의 평상시 수용인원의 쾌적성 확보뿐만 아니라 유사시에 유독가스의 원활한 배출을 통해 시민의 안전을 확보하는 것은 절대적으로 중요한 기술이라 할 수 있다. 이에 따라 다양한 화재 및 그에 따른 연기의 거동을 정확히 해석하고 예측할 수 있는 기술의 필요성에 대한 인식이 보편화되고 이에 대한 연구가 활발히 이루어져 왔으며, 앞으로도 실험 및 이론 연구, 수치계산을 통해 더욱 폭넓은 연구가 필요한 상황이다. 다양한 실험 기법의 발전에도 불구하고 실측 화재 실험에서의 과도한 비용 문제뿐만 아니라, 측정 상의 위험성 및 측정 장비의 한계로 인해서 실험연구는 많은 한계점을 보여주고 있다.

화재현상을 수치적으로 모사하기 위한 전산해석 기법은 크게 존(zone) 모델과 필드(field) 모델로 구분된다. 존 모델은 구획화재에 주로 사용되는 계산 기법으로, 구획 내부의 공간을 상부의 고온 연기층과 하부의 차가운 공기층으로 구분한다[1]. 최근 개선된 존 모델이 소개되어 예측 정확도는 부분적으로 개선되었으나, 제한된 화재 현상에만 사용되는 한계가 있다. 반면, 필드 모델 중 미국 NIST(National Institute of Standards and Technology)의 BFRF(Building and Fire Research Laboratory)에서 개발된 FDS는 LES(Large Eddy Simulation)기법을 적용하고 있어, 3차원 공간에서 매순간 화염의 거동 해석이 가능한 계산도구이다. 다양한 조건의 화재실험에 대한 검증이 지속적으로 수행되고 있어, 화재현상을 이해하는데 유용한 도구로 널리 알려져 있다. 현재는 버전 5까지 공개된 상태이며, 버전 6을 준비하는 과정에 있다[2]. 여기서 고질적인 한계였던 곡면 구현 문제를 해소하는 가상경계법이 적용된 바를 소스코드를 통해 확인할 수 있었다. 그러나 여전히 구(sphere)나 원형실린더(cylinder)에 국한된 곡면 구현만 가능한 상태임을 또한 확인할 수 있었다.

본 연구에서는 FDS6의 곡면 구현 기능을 확장하여 임의의 자유로운 곡면의 장애물을 구현할 수 있도록 표시입자(marker particle) 기법[3]을 FDS6에 도입하였다. 더불어 일반적인 설계 도구인 AutoCAD에서 설계된 임의 형상으로부터 형상정보를 읽어 들여, 이를 FDS6에서 구현하는 방법을 개발하였다. 이를

접수일: 2011년 8월 2일, 수정일: 2011년 10월 4일,

게재확정일: 2011년 10월 5일.

1 정회원, 인하대학교 대학원 기계공학과

2 정회원, 한국철도기술연구원

3 정회원, 인하대학교 기계공학과

* Corresponding author, E-mail: ksyang@inha.ac.kr

통해 화재 해석 및 일반 유동 해석시의 FDS의 활용성을 극대화하였다.

1.1 FDS의 지배방정식[4]

먼저 FDS를 구성하는 지배방정식을 살펴보기 위해, 질량 보존방정식, 운동량 보존 방정식, 에너지 보존 방정식 그리고 상태방정식을 다음 식 (1-5)에 각각 기술하였다.

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho \mathbf{u} = \dot{m}_b''' \quad (1)$$

$$\frac{\partial \rho Y_\alpha}{\partial t} + \nabla \cdot \rho Y_\alpha \mathbf{u} = \nabla \cdot \rho D_\alpha \nabla Y_\alpha + \dot{m}_\alpha''' + \dot{m}_{b,\alpha}''' \quad (2)$$

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) + \nabla p = \rho \mathbf{g} + \mathbf{f}_b + \nabla \cdot \boldsymbol{\tau}_{ij} + \mathbf{f}_{ibm} \quad (3)$$

$$\frac{\partial \rho h_s}{\partial t} + \nabla \cdot \rho h_s \mathbf{u} = \frac{Dp}{Dt} + \dot{q}''' + \dot{q}_b''' - \nabla \cdot \dot{\mathbf{q}}' + \varepsilon \quad (4)$$

$$p = \frac{\rho RT}{\overline{W}} \quad (5)$$

여기서, $\dot{m}_\alpha'''$ 는 단위 체적당 질량 생성율을 의미하며, α 는 화학종의 종류를 의미한다. Y_α 는 화학종 α 의 질량분율을 의미한다. 특히 $\dot{m}_b''' = \sum_\alpha \dot{m}_{b,\alpha}'''$ 는 스프링클러를 통해 계산 영역으로 유입되는 액체 입자에 의해 발생하는 화학종의 질량 생성율을 의미한다. 이와 마찬가지로 $\mathbf{f}_b$, $\dot{q}_b'''$ 는 액체 입자에 의한 운동량 부가항과 에너지 원천항을 각각 의미한다. $\mathbf{f}_{ibm}$ 는 가상경계법에 의하여 추가된 운동량 부가항을 의미한다. h_s 와 ε 는 각각 엔탈피와 에너지 소멸항이며, $\dot{\mathbf{q}}'$ 는 전도 및 복사 열유속을 뜻한다. $\dot{q}'''$ 는 화학반응으로부터 발생하는 단위체적당 열발출율을 의미한다. R 은 보편 기체상수를 $\overline{W}$ 는 혼합기체의 분자량을 각각 의미한다. 위 지배방정식은 직교좌표계에서 2차 정확도의 유한차분법으로 차분되며 특히 속도 벡터는 엇갈림 격자계를 사용한다. 시간에 대해서는 2차 정확도의 Runge-Kutta 방법으로 차분된다.

1.2 FDS6의 곡면 구현 방법 (SVN7573)[2]

앞 절의 운동량 방정식(식 (3))에서

1. n번째 압력 공간의 정수압 구배항, $\rho_n(z, t) \mathbf{g}$ 을 양변에서 빼주고, 정압 구배 관계식 $\nabla p = \rho_n \mathbf{g} + \nabla \tilde{p}$ ($\tilde{p}$ 는 섭동 압력)을 적용시킨 후,
2. 벡터 항등식 $(\mathbf{u} \cdot \nabla) \mathbf{u} = \nabla |\mathbf{u}|^2/2 - \mathbf{u} \times \boldsymbol{\omega}$ 를 적용하고,
3. 모든 항을 밀도 ρ 로 나누어 준 후,

4. 압력항을 다음과 같이 분리해 주고,

$$\frac{1}{\rho} \nabla \tilde{p} = \nabla \left(\frac{\tilde{p}}{\rho} \right) - \tilde{p} \nabla \left(\frac{1}{\rho} \right) \quad (6)$$

5. 가압력 H 를 $H = |\mathbf{u}|^2/2 + \tilde{p}/\rho$ 로 정의하고, 식 (3)을 식 (7)과 같이 나타낼 수 있다.

$$\begin{aligned} \frac{\partial \mathbf{u}}{\partial t} = & -(\mathbf{F} - \mathbf{f}_{ibm}/\rho + \nabla H) \\ \mathbf{F} = & -\mathbf{u} \times \boldsymbol{\omega} - \tilde{p} \nabla \left(\frac{1}{\rho} \right) - \frac{1}{\rho} \{ (\rho - \rho_n) \mathbf{g} + \mathbf{f}_b + \nabla \cdot \boldsymbol{\tau}_{ij} \} \end{aligned} \quad (7)$$

이후 FDS는 식 (7)에 다이버전스($\nabla \cdot$)를 취한 후 얻게 되는 H 에 대한 Poisson 방정식(식 (8))을 풀게 된다.

$$\nabla^2 H = -\frac{\partial}{\partial t} (\nabla \cdot \mathbf{u}) - \nabla \cdot (\mathbf{F} - \mathbf{f}_{ibm}/\rho) \quad (8)$$

고체면의 구현은 가상경계법을 적용하여 운동량 방정식에 다음 식과 같은 추가적인 운동량을 부가함으로써 지배방정식에 반영된다.

$$\mathbf{f}_{ibm} = \rho \left(\mathbf{F}^n + \frac{\mathbf{u}^{ibm} - \mathbf{u}^n}{\delta t} + \nabla H^n \right) \quad (9)$$

고체면 내부에서는 $\mathbf{u}^{ibm} = 0$ 로 설정함으로써 $\mathbf{u}^{n+1}$ 역시 0에 근사시킬 수 있다. 모든 장애물이 격자와 정확히 정렬된 상태라면, 이와 같이 쉽게 장애물을 구현하는 것이 가능하며, FDS 초기버전부터 FDS 5버전까지 사용되던 방법이다.

FDS 6버전부터는 곡면의 장애물을 구현할 수 있는 기법이 적용되었다[2]. 장애물의 외부, 즉 유체 부분에 위치한 속도 성분에 대해 경계층 방정식을 양해적으로 풀게 된다. 먼저 속도 성분이 정의된 위치에서 고체면에 대해 수선을 내린다. 수선의 발을 기준으로, 속도벡터 방향의 접선 방향(s)과 수직방향(n)이 새로운 2차원 직교 좌표계를 구성하게 되고, 이 새로운 좌표계에 좌표변환된 수직 속도 성분은 아래와 같이 계산하게 된다.

$$\overline{v^{k+1}} = \overline{v_w} + \delta n \left(\nabla \cdot \mathbf{u}^{k+1} - \frac{1}{2} \frac{\partial \overline{u}}{\partial s} \Big|_{2\delta n}^k \right) \quad (10)$$

그리고 접선방향 속도 성분은 운동량방정식으로부터 아래와 같이 차분된다.

Fig. 1 Grid and streamwise coordinate systems for the immersed boundary method implemented in FDS6 [2]

$$\frac{d\bar{u}}{dt} = - \left[\begin{aligned} & \bar{u} \frac{1}{2} \frac{\partial \bar{u}}{\partial s} \Big|_{2\delta n}^k + \bar{v}^{k+1} \frac{1}{2} \left(\frac{\partial \bar{u}}{\partial n} \Big|_{2\delta n}^k + \frac{\partial \bar{u}}{\partial n} \Big|_w \right) \\ & + \frac{1}{\rho} \left(\frac{\partial p}{\partial s} \Big|_{\delta n}^k + \frac{\tau_{sn} \Big|_{2\delta n}^k - \tau_{sn} \Big|_w}{2\delta n} \right) \end{aligned} \right] \quad (11)$$

여기서 $\bar{(\quad)}$ 는 좌표변환된 속도 성분을 의미하며 k 는 시간 단계, 하첨자 w 는 고체벽면에서 정의된 물리량을 의미한다. δn 은 고체면으로부터 속도 성분이 정의된 위치($\mathbf{x}_{velo}$)까지의 수직 거리를 의미한다. $\bar{\tau}_{sn}$ 은 $\mu \frac{\partial \bar{u}}{\partial n}$ 로 정의되는 전단응력 성분이다. Fig. 1은 FDS6에 적용되어 있는 곡면 구현 기법에 대한 이해를 돕는다.

1.3 현 FDS6의 곡면 구현 방법의 한계

FDS6에 곡면을 구현할 수 있는 가상경계법이 적용되었으나, 구현가능한 곡면의 형태는 구(sphere)와 원형실린더에 국한되어 있다. 앞 절에서 기술한 바와 같이, 체적력($\mathbf{F}^{ibm}$)이 부가될 속도 성분의 위치에서 곡면으로의 수선의 발을 형성하는 과정이 핵심적인 부분이다. 구(sphere)와 원형실린더 같은 형상은 수선을 내리는 것이 간단하다. 2차원 원의 중심과 반지름을 알고 있기 때문이다. 만약 방정식형태로 곡면이 정의 되었다면 반복법을 통해서라도 수선의 발 위치를 찾을 수 있다. 그러나 완전히 임의의 곡면에 대해 수선을 형성하는 것은 까다로운 과정이다. 따라서 본 연구에서는 표시입자(marker particles)를 도입하여 이를 해결하고자 한다.

2. 새롭게 적용된 수치 방법

2.1 표시입자(marker particle) 기법[3]

2차원 영역에서 임의의 형상을 가지는 물체의 경계를 방정식의 형태가 아닌 표시입자의 분포를 이용하여 구현하는 방법을 설명하고자 한다. 이는 3차원으로 확장 또한 용이하다.

Fig. 2 Arc length coordinate using marker particles

Fig. 3 A surface composed of (x_2, y_2) and the neighboring two points

표시입자의 분포를 이용하여 표면을 나타내기 위한 과정에서 가장 우선시 되어야 할 것은 아크길이(arc length)를 매개변수로 하여 표시입자의 분포를 표현하는 것이다. 우선, Fig. 2와 같이 임의 위치의 표시입자를 기준 입자로 선정한 후, 일정한 방향(시계방향, 혹은 반시계방향)으로 표시입자를 번호 붙인다. 임의의 2차원 물체는 표시입자들 사이를 직선 형태로 간주하여 형상화된다. 따라서 모든 표시입자는 기준점으로부터의 아크길이(arc length) s 값을 갖게 되고 이를 이용하면 표면정보를 매개변수화할 수 있다.

앞서 기술한대로 표시입자들 사이를 직선의 형태로 간주하여 아크길이(arc length)를 구하였으나, 표면의 형성과정에서는 2차 다항식 형태로 표현한다. 임의의 표시입자(s_i)를 기준으로 양 옆의 두 점(s_{i-1} , s_{i+1})을 이용하여 Fig. 3과 같이 2차 다항식을 구성하면 다음 식과 같다.

$$\begin{aligned} x(s) &= a_x s^2 + b_x s + c_x \\ y(s) &= a_y s^2 + b_y s + c_y \end{aligned} \quad (12)$$

여기서, 이미 알고 있는 각 표시입자의 좌표(x, y)와 아크길이(arc length) 좌표(s)를 이용하여 임의의 물체의 형상을 나타내는 다항식의 계수 $a_x, b_x, c_x, a_y, b_y, c_y$ 를 구할 수 있다. 즉, x 좌표에 대해서는 미지수 3개(a_x, b_x, c_x), 방정식의 개수 3개인 연립방정식의 해를 구하는 과정이 필요하게 된다. 이 때 크래머 공식(Cramer's rule)을 이용하였다.

2차 다항식으로 표면을 표현하는 방법을 택하고 있기 때문에 실제 물체표면이 3차 이상의 식으로 표현될만한 날카로운

Fig. 4 Three marker particles for the quadratic polynomial

모서리로 형성되었다면, 이는 2차 함수 형태로 수정 처리되는 썸이다. 찾고자하는 좌표의 s 가 s_{i-1} 과 s_i 사이의 값일 때, 더 가까운 점을 기준으로 3개의 표시입자를 선정하게 된다.

Fig. 4에서 ①의 위치와 같이 s_i 보다 s_{i-1} 에 좀 더 가까운 위치라면, x, y 값은 s_{i-1} 을 기준으로 구성된 2차식을 사용한다. 반면, ②와 같이 s_i 에 좀 더 가까운 위치라면, x, y 값은 s_i 을 기준으로 구성된 2차식을 사용하여 추정한다.

운동량을 추가하는 위치에서 임의의 고체표면에 수선의 발 (x_n, y_n) 을 형성하는 과정은 FDS에 적용된 IBM 기법에서 가장 중요한 요소이다(Fig. 5). 수선의 발은 운동량 부가 위치에서 고체표면으로의 최단거리에 해당하는 점좌표이다. 운동량 부가 위치에서 가장 가까운 표시입자의 아크길이 s 를 s_b , (x_n, y_n) 점의 s 값을 s_n 이라 두자. s_n 을 구하게 되면 식 (12)를 통해 (x_n, y_n) 점을 알 수 있기 때문에 최종 목표는 s_n 을 찾는 것이다. (x_n, y_n) 점에서의 단위 수직 벡터를 다음과 같이 표현할 수 있다.

$$\frac{x_i - x_n}{\sqrt{(x_i - x_n)^2 + (y_j - y_n)^2}} = \frac{-y_{s_n}}{\sqrt{(x_{s_n}^2 + y_{s_n}^2)}} \quad (13)$$

$$\frac{y_j - y_n}{\sqrt{(x_i - x_n)^2 + (y_j - y_n)^2}} = \frac{x_{s_n}}{\sqrt{(x_{s_n}^2 + y_{s_n}^2)}} \quad (14)$$

이 두 식을 한 식으로 표시하면 다음과 같다.

$$(x_i - x)x_{s_n} = -y_{s_n}(y_j - y_n) \quad (15)$$

이 때 x_{s_n}, y_{s_n} 는 각각 다음과 같다.

$$\begin{cases} x_{s_n} = 2a_x s_n + b_x \\ y_{s_n} = 2a_y s_n + b_y \end{cases} \quad (16)$$

식 (16)을 식 (15)에 대입하여 정리하면, 다음과 같다.

$$(x_i - x_n)(2a_x s_n + b_x) + (y_j - y_n)(2a_y s_n + b_y) = 0 \quad (17)$$

Fig. 5 Interface-normal point

구하려고 하는 미지수 s_n 는 식 (17)로부터 다음과 같이 구할 수 있다.

$$\begin{aligned} s_n &= \frac{-b_x(x_i - x_n) - b_y}{2a_x(x_i - x_n) + 2a_y(y_j - y_n)} \\ \begin{cases} x_n = a_x s_n^2 + b_x s_n + c_x \\ y_n = a_y s_n^2 + b_y s_n + c_y \end{cases} \end{aligned} \quad (18)$$

식 (18)을 s_n 의 다항식 형태로 표시하면 다음과 같다.

$$\begin{aligned} &(2a_x^2 + 2a_y^2)s_n^3 + (3a_x b_x + 3a_y b_y)s_n^2 \\ &+ (2a_x c_x + 2a_y c_y + b_x^2 + b_y^2 - 2a_x x_i - 2a_y y_i)s_n \\ &+ (b_x c_x + b_y c_y - b_x x_i - b_y y_i) = 0 \end{aligned} \quad (19)$$

결과적으로 s_n 에 대한 3차 방정식이 구성되며 이 방정식의 근을 구하기 위해 Newton-Raphson method를 사용하였다. 이 때, 초기값은 가장 가까운 표시입자 정보를 이용하였다.

3. 수정된 코드의 검증

기존 FDS의 소스코드(SVN7573)에 4900여 줄로 구성된 새로운 모듈(MARK.f90) 파일을 추가함으로써, 표시입자 방법을 적용하였다. 새로운 소스파일의 추가로 인해서, 소스파일 컴파일 시에 이를 반영해 주어야 한다. 현재 병렬계산은 고려되지 않았으며 단일 코어 계산만 가능하다.

3.1 구 주위 유동 해석 (Re=300)

Re=300의 구 주위 유동을 수정 전 FDS를 이용하여 계산을 수행하고 그 결과를 새롭게 수정된 FDS 코드를 이용해 계산한 결과와 비교하였다. 이미 FDS에 구현된 구 주위 유동을 구현하는 방법은 다음과 같다.

Fig. 6 Configuration for the calculation of flow past a sphere (SMokeyView, $Re=300$)

Fig. 7 Instantaneous velocity vectors and streamlines of the flow past a sphere in x-z plane at $Re=300$

```
&GEOM SHAPE='SPHERE', XYZ=2,4,4, RADIUS=0.5 /
```

XYZ 값은 구의 중심좌표를 의미하며, RADIUS 값은 구의 반지름을 의미한다. 두 계산의 정량적 비교를 위해 아래와 같이 항력과 양력을 계산하였다.

```
&DEVC ID='F_X', XB=1.25, 2.75, 3.25, 4.75, 3.25, 4.75
      QUANTITY='F_X'
      STATISTICS='TENSOR SURFACE INTEGRAL/'
&DEVC ID='F_Z', XB=1.25, 2.75, 3.25, 4.75, 3.25, 4.75
      QUANTITY='F_Z'
      STATISTICS='TENSOR SURFACE INTEGRAL/'
```

위에서 보듯, 물체 표면에서 적분된 유체력을 계산하는 것이 아닌, XB로 표현되는 구를 넉넉히 포함한 직육면체 표면에서의 유체력을 계산하는 방법이 현재 FDS에 구현되어있는 유체력 계산 방법이다.

계산의 영역은 Fig. 6에 도시한 바와 같이 $8m \times 8m \times 8m$ 로 설정하였다. $Re=300$ 의 구 주위 유동은 비대칭 비정상상태 유동으로 알려져 있다[5-7]. 임의의 시간에서의 순간 유동장을 Fig. 7에 도시하였으며, x-z 평면에서의 유선 및 속도벡터의 양상

Fig. 8 Variations of drag coefficients : (a) conventional FDS, (b) modified FDS using marker particles

Fig. 9 Configuration of the flow past an arbitrary object : inlet velocity (U)= $0.036m/s$, $l=0.25m$, $r=0.03m$, $Re=Ul/\nu=600$

을 통해 그 비대칭성을 확인할 수 있다. 이렇게 계산된 구의 항력계수를 시간에 따라 도시하여 Fig. 8에 나타내었다. FDS에서는 임의 체적이 받는 유체력에 대한 반력을 계산해주기 때문에 그림 상에는 음의 부호로 표시되는 걸 확인할 수 있다. Fig. 8(a)는 기존 FDS로 계산 수행한 결과이며, Fig. 8(b)는 표시입자가 적용된 FDS로 계산을 수행한 결과이다. 그림에서 보듯, 두 결과가 거의 일치하는 것을 알 수 있다. (a)의 경우, 평균 항력계수가 0.829이며, (b)의 경우는, 평균 항력계수가 0.827이다. 이를 통해 얻은 스트루할수(St)는 두 경우 모두 0.131로 완전히 동일한 것도 확인할 수 있었다.

4. 응용 예제 계산 결과

4.1 뭉툭한 원뿔 주위 유동 ($Re \equiv Ul/\nu=600$)

기존의 FDS에서는 구형이 불가능한 임의 형상의 곡면 장

Fig. 10 Instantaneous velocity and pressure fields at run time=1000sec in x-y plane plotted by SMV

Fig. 11 Instantaneous velocity field at run time=1000sec in x-z plane posted by Tecplot

애몰이 정확히 구현되는 지를 평가하기 위해 Fig. 9에 도시한 바와 같은 모서리가 곡면처리된 원뿔을 표시입자로 구현하여 계산을 수행하였다. 아크길이(arc length)를 정의하기 위해서는 교차 표면 위에 2차원 메시(mesh) 위에 표시입자를 위치시켜야만 한다. 본 검증에 사용된 형상에서는 48×108 개의 표시입자를 사용하였다. Fig. 9의 좌측 그림에서 화살표는 유동의 방향을 의미한다. 계산영역은 x, y, z 방향으로 $2m \times 1m \times 1m$ 를 설정하였으며, 장애물 주위에서는 격자크기 $7.8125mm$, 장애물에서 멀리 떨어진 곳에서는 $15.625mm$ 크기의 균일한 격자를 사용하였다. 사용된 총 격자수는 약 75만개이다.

Fig. 10은 FDS의 전용 후처리프로그램인 SMOkeView를 통해 run time 1000초에서 장애물의 중앙을 관통하는 x-y 평면에서의 속도장과 압력장을 도시한 그림이다. 입의 형상을 입체적으로 표시해 줄 수 있는 SMOkeView의 소스코드까지는 수정하지 않은 상태이기 때문에 명확한 장애물의 위치를 확인할 수는 없지만, 원하는 장애물이 구현되고 그 주위 유동이 물리적으로 타당함을 확인할 수 있다. Fig. 11은 FDS의 출력

Fig. 12 A 3DSOLID entity and A PLANESURFACE entity

파일에서 데이터를 추출하여 후처리 소프트웨어인 Tecplot을 통해 x-z 평면에서의 속도장을 도시한 그림이다. Fig. 10과 Fig. 11을 통해 표시입자를 이용한 가상경계법이 성공적으로 FDS에 접목된 것을 확인할 수 있다.

5. CAD에서의 표시입자 정보 추출

위에서 FDS에 적용한 표시입자 기술의 활용성을 극대화하기 위해서는 일반적인 CAD 프로그램에서 설계된 형상을 FDS로 옮겨오는 기술이 필수적이다. 본 장에서는 AutoCAD 2012 버전에 AutoLISP 언어를 통해 이를 구현한 과정을 설명하고자 한다. AutoLISP는 널리 사용되는 AutoCAD의 기능을 확장하고 사용자화하기 위해 기존의 XLISP의 초기버전을 수정해서 Autodesk사에서 만들어 낸 개발도구이며 기존의 표준 LISP 함수를 외에도 복잡한 AutoCAD 연산기능들을 처리할 수 있는 강력하고 다양한 함수들을 제공한다. 본 연구에서 개발된 알고리즘을 담은 AutoLISP 파일은 AutoCAD에서 손쉽게 실행가능하다. 해당 알고리즘을 순차적으로 정리하면 다음과 같다.

5.1 3DSOLID 모델과 절단면(PLANESURFACE) 구성

구현하고자 하는 3차원 형상을 반드시 3DSOLID 속성으로 모델링한다. 그리고 z 방향 절단면을 구성하게 될 1개의 x-y 방향 평면(PLANESURFACE)을 모델링한다. 이때, 평면은 3차원 형상의 z 방향 가장 낮은 위치를 관통하도록 모델링한다. 이는 다음 단계에서 3DSOLID 모델에 절단면을 형성하는 기준면으로 활용된다(Fig. 12).

5.2 3DSOLID 모델 절단면 형성 (interfere)

x-y 방향 평면(PLANESURFACE)을 양의 z 방향으로 입력받은 크기만큼씩 이동시켜가면서 PLANESURFACE와 3DSOLID 간의 교차면을 생성하게 된다. 이때, AutoCAD 내의 interfere 명령어가 사용된다. 이러한 과정은 미리 입력받은 횟수만큼

Fig. 13 Interfered surfaces between 3DSOLID and PLANESURFACES

Fig. 14 Joined POLYLINE entities

반복된다. 결과적으로 SURFACE 속성을 갖는 다수의 절단면이 형성된다(Fig. 13). 이때, 절단면은 복수를 허용하지 않는다. 즉, 현재의 버전에서는 너무 굴곡이 심한 형상은 구현이 불가능하다. 단, 굴곡이 심하더라도 z 방향 축을 복수의 절단면이 생성되지 않도록 설정한다면 정상적인 구현이 가능하다.

5.3 절단면의 모서리 추출 (xedge)

이렇게 형성된 절단면들을 선택한 후, 모서리 추출 기능(xedge)을 사용하면 교차면의 테두리가 단위 개체들로 새롭게 형성된다. 이때, 절단면의 형상정보에 따라, 선(LINE), 다중선(POLYLINE, LWPOLYLINE), 스플라인선(SPLINE), 타원(ELLIPSE), 원(CIRCLE) 등의 요소들이 형성된다.

5.4 동일 절단면 상의 선 요소들 결합 (pedit)

한 개의 절단면 가장자리의 선 요소들을 하나의 다중선(POLYLINE)으로 묶어야만, 다음 단계의 등간격 점좌표 형성이 가능하게 된다. 다양한 속성의 선 요소들을 하나로 묶기 위해서는 모두 다중선(POLYLINE) 속성으로 변형시켜야 한다. 이러한 일괄적인 작업을 위해, 미소크기만큼의 오프셋(offset) 명령을 실행하게 되는데, 타원이나 스플라인선 등이 다중선 속성으로 변형된다. 단일한 다중선으로의 결합을 위해서, 모든 다중선 요소에 explode 명령을 적용하여 최소 단위의 선 요소로 분해한 후 pedit 명령어를 통해 재결합 과정을 진행한

Fig. 15 Marker particles generated

Fig. 16 Structure of marker particles plotted by TECPLOT

다. 이렇게 결합이 완료되면, 앞서 생성됐던 교차면의 개수와 동일한 수의 다중선(POLYLINE) 요소가 도면에 존재하게 된다(Fig. 14).

5.5 다중선의 등간격 분리 (divide)

이제 각 다중선을 등간격으로 분리하여 좌표값을 표시하게 되면, 그 점좌표들이 바로 FDS에서 사용하게 될 표시입자들의 좌표값이 된다. divide 명령을 통해 각각의 다중선 상에 등간격으로 점 요소들이 생성된다(Fig. 15).

5.6 점 좌표의 정렬

수정된 FDS 코드에서는 점좌표의 정보를 읽어 들이는 순서가 매우 중요하다. 따라서 AutoCAD에서 산발적으로 읽어 들인 점좌표를 I, J 방향의 구조화된 순서로 정렬할 필요가 있다.

5.7 뒤틀린 구조를 방지하기 위한 재정렬

인접한 절단면의 모양이 급격히 변형되거나 면적이 급격히 변화하는 구간에서는 점 좌표들의 정렬 구조가 심하게 뒤틀리는 현상이 발생할 수 있다. 이는 수정된 FDS가 구동될 때 문제를 발생시킬 수 있다. 따라서 뒤틀림이 최소화되도록 형상의 표면정보가 크게 변형되지 않는 범위 내에서 점좌표를 새롭게 정의하여 정렬하게 된다.

Fig. 17 Simply-designed train in AutoCAD2012

(a)

(b)

Fig. 18 Results of FDS calculation for flow past a simply-designed train : (a) velocity vectors colored by pressure in z-x plane (SMV), (b) contours of the magnitude of streamwise(z) velocity (TECPLOT)

5.8 파일에 쓰기

수정된 FDS에서 표시입자 정보를 읽어 들일 수 있도록 점 좌표를 파일에 기록한다. 더불어, TECPLOT 용 후처리 파일도 출력한다. 3차원 포맷으로 불러 들여 SHADE 처리를 해주면, Fig. 16과 같이 표시입자의 표면 구현을 확인할 수 있다. 본격적인 FDS 계산 전에 정상적인 좌표 구현 여부를 확인하는데 유용하게 쓰일 수 있다.

6. 검증용 계산 예 (고속전철)

검증 계산을 위해, AutoCAD2012에서 고속전철의 전두부를 대략적으로 모델하였다. 수직높이는 약 1m이며 주유동방향

Fig. 19 Force coefficients of the train

길이는 약 3m 정도이다. 주유동방향(z)으로의 절단면은 총 133개이며 각 절단면의 모서리를 따라 80개의 표시입자를 추출하였다. AutoCAD에서 도식된 모델링의 형상을 Fig. 17에서 확인할 수 있다. 계산 영역은 x, y, z 방향으로 각각 8m, 8m, 16m로 설정하였으며 장애물 주위 최소 격자 크기는 0.05m로 설정하였다. 입구속도는 18km/hour 정도이며, 장애물의 수직높이 기준의 Re는 약 330,000에 해당한다.

이로부터 표시입자의 정보를 추출하여 새로운 FDS에서 계산을 수행한 결과를 Fig. 18에 도식하였다. 그림 (a)는 압력분포로 채색된 z-x 평면에서의 속도벡터를 FDS 전용 후처리기인 SMOkeView에서 도식한 그림이다. SMOkeView에 임의의 장애물을 구현하는 것은 수정된 사항이 아니므로 고속전철 형상이 가시화되어 있지 않은 상태이며 장애물 내부의 압력은 의미 없는 값이므로 무시해도 좋다. 고속전철의 전두부의 압력이 상승한 것을 확인할 수 있음과 동시에 임의의 형상이 정확히 구현된 것을 확인할 수 있다. 그림 (b)에서는 수정된 FDS에서 출력된 PLOT3D 파일로부터 TECPLOT용 데이터 파일을 생성하여 도식한 그림으로서, 주유동방향 속도의 크기 분포를 색깔로 도식한 그림이다. 고속전철 후류에서의 복잡한 와류를 확인할 수 있으며 고속전철의 형상이 이상없이 구현된 것도 확인할 수 있다. Fig. 19는 고속전철이 받는 유체력을 시간에 따라 도식한 그림이다. 이는 기존의 FDS에서 사용하는 박스형태의 검사체적에서 계산된 유체력을 도식한 것이 아닌, 장애물 표면에서 직접 표면력을 적분하는 수정된 방법으로 계산된 유체력을 도식한 그림이다. 0.2~0.3 정도의 항력계수를 확인할 수 있으며, 전체적으로 양의 양력(Cl)을 받고 있는 것도 확인할 수 있다.

이로써, AutoCAD로부터 표시입자의 좌표정보를 읽어 들이는 알고리즘이 정상적으로 작동하는 것을 확인할 수 있다.

7. 결 론

연소 및 소방방재 분야에서 널리 사용되고 있는 필드모델

모사 도구인 FDS의 기능적인 개선을 도모하여 FDS의 학계 및 산업계에서의 활용가치를 극대화하였다.

현재 일반 공개를 준비 중인 FDS6 버전에서 부분적으로 구현된 가상경계법의 곡면 구현 기능을 더욱 일반화시켜 자유로운 곡면이 구현 가능하도록 수정하였다.

자유로운 곡면의 구현을 좀 더 용이하게 적용할 수 있도록, 일반적인 제도도구인 AutoCAD에서 표시입자 정보를 추출하는 알고리즘을 개발하였으며, 이는 AutoCAD 내에서 손쉽게 실행이 가능하다. 이로써, FDS를 이용한 임의 곡면 물체의 구현이 가능하게 되었다.

현재까지는 병렬계산에 대한 구동 가능 여부는 테스트되지 않았으며, 물체 표면에서의 열 및 화학 반응도 고려되지 않은 상태이다. 또한, 향후 이러한 부분이 보완된다면, FDS의 활용도 극대화에 크게 기여할 수 있을 것이다.

후 기

본 연구는 한국철도기술연구원의 도시철도 시설물 표준화 연구 사업(SW11002)의 지원에 의하여 수행되었음.

참고문헌

- [1] 2003, Olenick, S.M. and Carpenter, D.J., "An Updated International Survey of Computer Models for Fire and Smoke," *Journal of Fire Protection Engineering*, Vol.13, No.2, pp.87-110.
- [2] 2010, McDermott, R.J., Forney, G.P. McGrattan, K. and Mell, W.E., "Toward FDS 6: Complex geometry, embedded meshes, and quality assessment," *V European Conference on Computational Fluid Dynamics*, ECCOMAS CFD, Lisbon, Portugal, pp.14-17 June.
- [3] 2006, Yang, J. and Balaras, E., "An embedded-boundary formulation for large-eddy simulation of turbulent flows interacting with moving boundaries," *Journal of Computational Physics*, Vol.215, pp.12-40.
- [4] <http://fire.nist.gov/fds>.
- [5] 1997, Mohd-Yosuf, J., "Combined immersed boundary / B-spline methods for simulation of flow in complex geometries," *Annual Research Briefs*, Center for Turbulence Research, pp.317-328.
- [6] 1999, Johnson, T.A. and Patel, V.C., "Flow past a sphere up to a Reynolds number of 300," *Journal of Fluid Mechanics*, Vol.378, pp.19-70.
- [7] 2003, Constantinescu, G., Chapelet M. and Squires K., "Turbulence modeling applied to flow over a sphere," *AIAA Journal*, Vol.41, No.9, pp.1733-1742.

[1] 2003, Olenick, S.M. and Carpenter, D.J., "An Updated