EDISON_CFD

EDucation-research Integration through Simulation On the Net

제 6 강

시간 차분법 – Part I

박동훈
부산대학교 항공우주공학과

시간차분의 개념

- 시간에 대한 미분항을 포함하는 지배방정식
  - 시간과 공간에 대한 차분(discretization) 필요

$$\frac{\partial \phi}{\partial t} = F(\phi)$$

ϕ : scalar quantity

시간 미분항을 제외한 모든 항

- 시간 차분 (temporal discretization)
  - 1차 정확도 후방 차분법:

$$\frac{\phi^{n+1} - \phi^{n}}{\Delta t} = F(\phi)$$

- n-1, n, n+1 : Time Level
- Δt : Time Step

  - 2차 정확도 후방 차분법:

$$\frac{3\phi^{n+1} - 4\phi^{n} + \phi^{n-1}}{2\Delta t} = F(\phi)$$

  - 현재 및 이전 시간의 정보를 사용하여 다음 시간 값 계산

시간차분법의 분류

- 외재적 방법 (Explicit Method)
  - 현재 시간 단계 값(ϕ^n)을 이용하여 우변 결정

$$\frac{\phi^{n+1}-\phi^n}{\Delta t}=F\left(\phi^n\right) \quad \Rightarrow \quad \phi^{n+1}=\phi^n+\Delta t F\left(\phi^n\right)$$

- 내재적 방법 (Implicit Method)
  - 다음 시간 단계 값 (ϕ^{n+1})을 이용하여 우변 결정

$$\frac{\phi^{n+1}-\phi^n}{\Delta t}=F\left(\phi^{n+1}\right) \quad \Rightarrow \quad \phi^{n+1}-\Delta t F\left(\phi^{n+1}\right)=\phi^n$$

Example

$$\frac{\partial \phi}{\partial t}=c\frac{\partial \phi}{\partial x}$$

중앙차분 적용

$$\frac{\phi_i^{n+1}-\phi_i^n}{\Delta t}=c\frac{\phi_{i+1}^{n+1}-\phi_{i-1}^{n+1}}{2\Delta x}$$

i-1, i, i+1 세 위치에서의 n+1 시간단계 값 포함

$$\phi_i^{n+1}-c\frac{\Delta t}{2\Delta x}\left(\phi_{i+1}^{n+1}-\phi_{i-1}^{n+1}\right)=\phi_i^n$$

모든 위치에서의 차분식들로 연립방정식 계 구성

시간차분법의 분류

- 복합적 방법
  - 현재와 다음 시간단계 값의 조합으로 우변 결정

$$\frac{\phi^{n+1}-\phi^{n}}{\Delta t}=F\left(\phi^{*}\right) \qquad \phi^{*}=\theta\phi^{n+1}+\left(1-\theta\right)\phi^{n}$$

$$\frac{\phi^{n+1}-\phi^{n}}{\Delta t}=F\left(\left[\theta\phi^{n+1}+\left(1-\theta\right)\phi^{n}\right]\right)$$

  - 예: Crank-Nicolson Method θ=1/2

$$\frac{\phi^{n+1}-\phi^{n}}{\Delta t}=F\left(\frac{\phi^{n+1}+\phi^{n}}{2}\right)$$

시간 정확도

- 시간 정확도 (Time Accuracy)
  - 시간 차분에 따른 차분 오차 발생

1차 정확도 시간 차분법 오차 ~ $O(\Delta t)$

2차 정확도 시간 차분법 오차 ~ $O\left((\Delta t)^2\right)$

  - 높은 차수의 방법일수록 시간 차분 오차 감소
  - time step이 클 수록 시간 차분 오차 증가
  - 각 시간 단계에서의 결과 정확도에 영향
  - 시간 전진에 따라 오차 누적 가능 – 긴 시간 계산 시 오차 증가

외재적 시간 차분법 (1)

- Euler Backward Difference

$$\frac{\phi^{n+1}-\phi^{n}}{\Delta t}=F\left(\phi^{n}\right)$$

1차 정확도

$$\frac{3\phi^{n+1}-4\phi^{n}+\phi^{n-1}}{2\Delta t}=F\left(\phi^{n}\right)$$

2차 정확도

  - 높은 차수의 시간 정확도를 위해서는 더 많은 이전 시간 단계에서의 값들이 필요
  - 필요한 이전 시간단계들의 정보를 저장하고 있어야 함
  - 정확도에 비례하여 요구 메모리 증가

외재적 시간 차분법 (2)

- Multi-Stage Runge-Kutta Scheme
  - 여러 중간 단계를 거쳐 다음 시간 값을 예측

m-stage R-K method

$$\Delta\phi = \phi - \phi^n \qquad \phi^{(0)} = \phi^n$$

Stage 1: $\Delta\phi^{(1)} = \alpha_1 \Delta t\, R\left(\phi^{(0)}\right)$ $\qquad \phi^{(1)} = \phi^{(0)} + \Delta\phi^{(1)}$

Stage 2: $\Delta\phi^{(2)} = \alpha_2 \Delta t\, R\left(\phi^{(1)}\right)$ $\qquad \phi^{(2)} = \phi^{(0)} + \Delta\phi^{(2)}$

$\vdots$ $\qquad\qquad$ $\vdots$ $\qquad\qquad$ $\vdots$

Stage m: $\Delta\phi^{(m)} = \alpha_m \Delta t\, R\left(\phi^{(3)}\right)$ $\qquad \phi^{(m)} = \phi^{(0)} + \Delta\phi^{(m)} = \phi^{n+1}$

α_i : multi-stage coefficient for i^{th} stage

  - 시간정확도와 단계(stage)수에 따라 계수가 달라짐
  - 같은 시간정확도 일 때 단계수가 많을수록 긴 time step 사용 가능

외재적 시간 차분법 (3)

- Example : 4-Stage Runge-Kutta Scheme

$$\Delta\phi = \phi - \phi^n \qquad \phi^{(0)} = \phi^n$$

Stage 1: $\phi^{(1)} = \phi^{(0)} + \alpha_1 \Delta t\, R\left(\phi^{(0)}\right)$ Stage 2: $\phi^{(2)} = \phi^{(0)} + \alpha_2 \Delta t\, R\left(\phi^{(1)}\right)$

Stage 3: $\phi^{(3)} = \phi^{(0)} + \alpha_3 \Delta t\, R\left(\phi^{(2)}\right)$ Stage 4: $\phi^{(4)} = \phi^{(0)} + \alpha_4 \Delta t\, R\left(\phi^{(3)}\right)$

$$\phi^{(4)} = \phi^{n+1}$$

- Modified Runge-Kutta Method
  - CFD에서 multi-stage Runge-Kutta 방법의 적용 : 매 단계에서 새로운 예측값을 사용하여 점성항 또는 소산항 재평가 → 계산량 증가
  - 매 단계에서 새로 평가하지 않고 다른 단계에서의 값을 조합하여 사용 → 계산 효율 및 수치적 안정성 증대 가능

외재적 방법의 특성

- 수치적 안정성 (numerical stability) 문제
  - 사용 가능한 time step (Δt)이 수치적 안정성에 의해 제약 받음
  - 수치적 안정성 확보를 위해 특정 크기 이하의 Δt 사용

- CFL (Courant-Friedrichs-Lewy) Condition
  - 수치적 안정성을 위한 필요조건

Example: 1차원 문제

Courant number: $C = \dfrac{u\Delta t}{\Delta x}$

- u : magnitude of velocity
- Δx : the length interval

CFL condition: $C \le C_{\max}$

  - CFL 조건을 만족하는 Δt 를 사용해야 함 (속도, 격자크기가 영향을 줌)
  - 다수의 CFD 코드에서 CFL number를 사용하여 Δt 설정

내재적 시간 차분법 (1)

- ADI (Alternating Direction Implicit) Scheme
  - 한 방향으로의 내재적 계산을 각 방향에 대해 순차적으로 수행
  - 차분식의 근사 분해(Approximate Factorization)

$$\left(1+D_xA+D_yB\right)\Delta\phi = R\left(\phi^n\right)$$

x, y 방향 공간 미분 연산자

근사분해

$$\left(1+D_xA\right)\boxed{\left(1+D_yB\right)\Delta\phi}^{\Delta\phi^*} = R\left(\phi^n\right)$$

$\left(D_xA\right)\left(D_yB\right)\Delta\phi$: factorization error

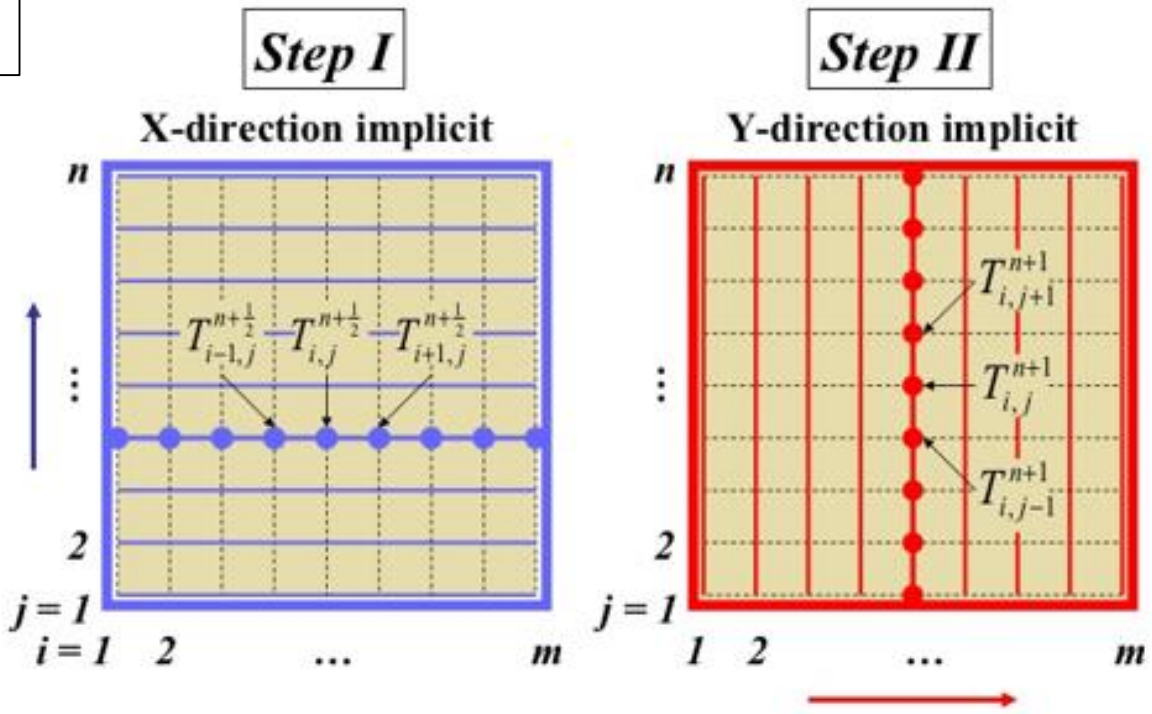
Step 1: $\left(1+D_xA\right)\Delta\phi^* = R\left(\phi^n\right)$ — x 방향 implicit → $\Delta\phi^*$ 획득

Step 2: $\left(1+D_yB\right)\Delta\phi = \Delta\phi^*$ — y 방향 implicit → $\Delta\phi$ 획득

내재적 시간 차분법 (2)

Concept of ADI

$$\frac{\partial T}{\partial t} = k\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\right)$$

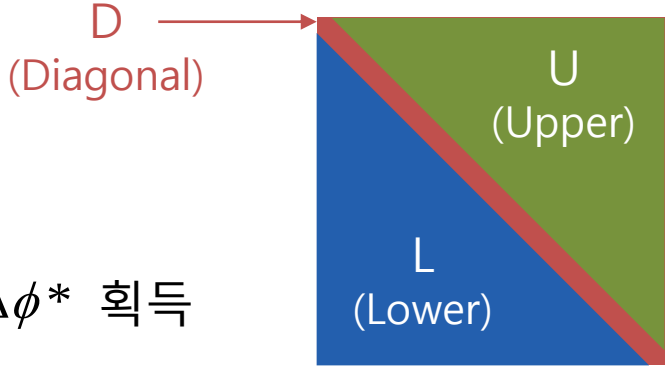


- AF-ADI (Alternating Direction Implicit) Scheme in CFD
  - 정렬격자(structured grid)에만 적용 가능
  - Factorization Error 추가 / 병렬화 이슈

내재적 시간 차분법 (3)

- LU-SGS (Lower-Upper Symmetric Gauss-Seidel) Scheme
  - Gauss-Seidel relaxation 방식 적용 / 전방 & 후방 Sweep 수행

$$(D+L)\underbrace{D^{-1}(D+U)\Delta\phi^n}_{\Delta\phi^*} = R(\phi^n)$$



Forward sweep: $(D+L)\Delta\phi^* = R(\phi^n)$ → $\Delta\phi^*$ 획득

Backward sweep: $(D+U)\Delta\phi^n = D\Delta\phi^*$ → $\Delta\phi$ 획득

  - 정렬 격자와 비정렬 격자에 모두 사용 가능
  - 수치적 복잡성과 요구 메모리량이 다단 R-K 방법과 비슷한 수준
  - 상대적으로 벡터 컴퓨팅과 병렬화 적용에 용이함

내재적 방법의 특성

- 높은 수치적 안정성
  - Fully implicit 방법은 unconditionally stable
  - Time step에 대한 제약 완화
- 추가적인 계산 요구
  - 한번의 시간 전진에 더 많은 (추가적인) 계산 요구 – 계산량 증가
- 구현 및 적용성
  - 선형화 등을 비롯하여 적용 및 구현이 상대적으로 복잡함

시간 차분법 I - 요약

- 현재 시간단계(n) 값으로부터 다음 시간레벨(n+1) 계산

	Explicit Method	Implicit Method
Advantage	× 상대적으로 쉬운 적용 × 시간 정확도 차수 증가 용이	× 수치적 안정성 × Time step 제약 완화
Disadvantage	× CFL 조건 - 수치적 불안정성 × Time step 제약	× 추가계산 및 적용 복잡성
Methods	× Euler Backward Difference × Multi-stage Runge-Kutta	× AF-ADI × LU-SGS