

정렬 격자 기반의 효율적인 셀단위 적응 격자 알고리즘 개발 연구

Development of an Efficient Cell-by-Cell Adaptive Mesh Refinement Algorithm on Structured Grids

윤성환^{1*}, 김종암¹
서울대학교 기계항공공학부¹

초 록

정렬 격자 기반의 효율적인 적응 격자 기법으로서, 새로운 형태의 셀단위 적응 격자 세밀화 알고리즘을 개발하였다. 새로이 개발된 셀단위 세밀화 기법의 핵심 아이디어는 주로 중첩 격자 기법에서 사용되는 격자의 분류 방법을 도입하는 것이다. 분류된 격자 형태를 사용함으로써, 본 연구에서 개발한 셀단위 세밀화 알고리즘은 격자 시스템의 구조화된 인덱스를 유지하면서도 불필요한 격자점들의 세밀화를 막아 기존 기법에 비해 계산 시간 및 메모리 측면에서 효율적이다. 수치 실험을 통하여 제안된 기법의 정확성 및 효율성을 테스트해 보았으며, 적응 격자 기법을 사용하지 않은 경우에 비해 정확성과 효율성이 월등함을 확인할 수 있었다.

ABSTRACT

As an efficient AMR method on structured grids, a new cell-by-cell refinement algorithm is developed. Key idea of the new cell-by-cell refinement is to use categorized cell types which are originated from overset mesh technique. Thanks to categorized cell types, present method provides efficient cell-by-cell refinement algorithm, while preserving structured index of grid system. Through numerical test cases, the performance of present algorithm is investigated and comparison of accuracy and efficiency shows that present method is much more accurate and efficient than the case without AMR method.

Key Words : Adaptive Mesh Refinement (적응 격자 기법), Structured AMR (정렬 적응 격자 기법), Cell-by-Cell AMR (셀단위 적응 격자 기법), h -refinement (h -세밀화)

1. 서 론

Euler/Navier-Stokes 방정식을 이용한 유동 해석은 이론적으로는 강한 충격파 및 와류의 전이 과정을 정확하게 모사할 수 있으나, 수치 해석을 통해 이를 구현할 때에는 반드시 정확한 예측이 가능한 것은 아니다. 과도한 수치 점성 및 불충분한 격자수로 인해, 와류의 전달 과정에서 예측하고자 하는 와류가 급속히 소멸되는 현상은 이전 연구에서 많은 연구자들이 경험한 바 있는 사실이다. 이러한 수치적 소산은 공력 계수 예측, BVI 소음 예측 등 많은 부분에서 유동 해석의 정확성을 떨어뜨리는 요인이 된다.

일반적으로 Euler/Navier-Stokes 방정식을 이용한

수치적 유동 해석의 정확도와 효율성을 향상시키는 방법으로는 몇 가지 서로 다른 방법이 존재하며, h - (격자 크기), p - (내삽 기법의 차수), hp - 및 r - (격자 재배치) refinement 가 대표적이다. h -refinement 기법은 내삽 기법의 정확도는 고정하고 격자의 세분화를 통해 정확도를 향상시키는 방법이며, p -refinement 기법은 고정된 격자수에서 내삽 기법의 차수를 증가시킴으로써 해의 정확도를 향상시키는 방법이다. 그리고 hp -refinement 기법은 h -기법과 p -기법을 결합한 방법이며, r -refinement 기법은 격자의 재배치를 통하여 유동 해석의 정확도를 향상시킨다. 본 연구에서는 h -refinement 를 이용하여 유동 해석의 정확도와 효율성을 향상시키는 방법에 대하여 살펴보고자 한다.

2. 정렬 격자 기반의 적응 격자 기법

유동해에 기반한 적응 격자 기법은 주로 Cartesian 격자계 및 비정렬 격자계에 대하여 활발히 연구되어 왔다⁽¹⁾. 정렬 격자계를 기반으로 적응 격자 기법을 적용한 연구는 비정렬 격자계에 비해 상대적으로 적은데, 이는 정렬 격자계의 특성상 필요로 하는 위치에 새로이 격자를 추가하는 것이 어려운 경우가 많기 때문이다. 그러나, 많은 연구자들의 노력한 결과 몇몇 주요한 기법들이 제안되었으며 다양한 형태로 발전되었다⁽²⁾. 이 중 본 연구와 관련된 기법 및 각 기법의 특징에 대해 살펴보기로 한다.

2.1 Block-by-block AMR algorithm

이 기법은 M. J. Berger에 의해 최초로 제안되었으며^(3, 4), 정렬 격자 기반의 적응 격자 기법으로써 현재까지도 가장 널리 사용되어지는 방법이다.

기본적으로 이 알고리즘에서는 세밀한 격자를 가지는 영역이 몇 개의 직사각형 격자 패치로 구성된다. 즉, 기본 격자에서 계산된 에러가 일정 기준을 초과함으로써 격자 세밀화가 필요하다고 판단된 격자점들이 이러한 직사각형 하위 격자계에 모두 포함된다.

이에 대해 좀 더 살펴보자면, 격자 계층에 대하여 일련의 레벨을 정하고($l=1, \dots, l_{max}$), 기본 격자계 즉, 가장 넓은 격자 간격을 가지는 레벨을 $l=1$ 로 하여 격자 간격이 세밀함에 따라 레벨이 높아지도록 정의한다. 하나의 레벨에 대해서는 여러 개의 직사각형 패치가 존재할 수 있으며, 이보다 높은 레벨의 격자 패치를 모두 포함하여야 한다 (Fig. 1. 참고). Fig. 2.는 실제 계산에 적용된 사례로, 여러 개의 직사각형 패치 내부는 세밀한 격자들로 구성된다.

직사각형 하위 격자계를 사용하여 기본 격자계와 하위 격자계를 분리함으로써 기본 격자에서 사용되는 것과 동일한 수치 기법을 하위 격자계에도 적용할 수 있다. 따라서 이 기법은 코드에 적용하기에 편리하며, 벡터 및 병렬 컴퓨팅 시 계산 효율성이 굉장히 뛰어나다.

2.2 Cell-by-cell AMR algorithm

앞서 설명한 M. J. Berger의 방법^(3, 4)에서는 격자 세밀화 과정이 직사각형 격자 패치 즉, 블록 단위로만 진행되어진다. 블록 단위로 적응 격자를 적용하는 방법의 가장 큰 단점은 불필요한 격자점

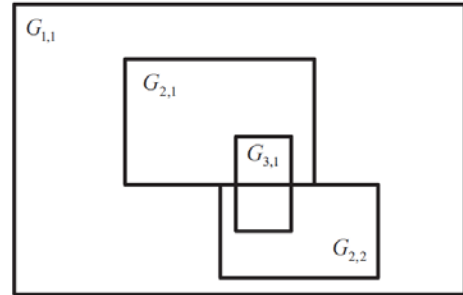


Fig. 1. An example of grid hierarchy for block-by-block AMR algorithm⁽⁴⁾

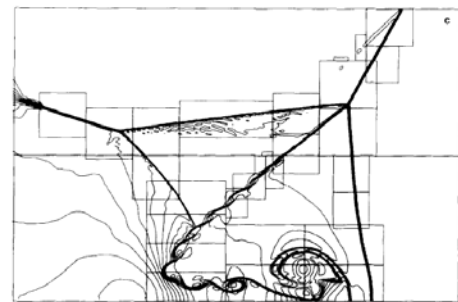


Fig. 2. An example of block-by-block AMR⁽⁴⁾ for double Mach reflection problem

에 대해서도 격자 세밀화가 이루어진다는 것이다. 격자 세밀화 여부를 판단하기 위하여 일정 기준을 정하고 기준에 부합되어 세밀화된 격자점들이 정해지고 나면, 블록 단위로 그 격자점들을 포함하여 세밀화를 수행한다. 이때 세밀화가 필요없다고 판단된 일부 격자점들도 블록 내에 포함될 수 밖에 없다.

세밀화가 필요 없다고 판단된, 즉, 충분히 정확한 해를 도출하고 있는 격자점에 대해서도 격자 세밀화를 수행함으로써 결국 계산 효율성이 저하되는 영향을 미칠 수 있다. 또한, 불필요한 격자 세밀화의 수행을 줄이기 위해서는 Fig. 2.에서 볼 수 있듯이 많은 수의 블록을 사용하여야 하는데 이 과정에서 발생하는 복잡한 블록간 계층 구조는 코드의 복잡성을 증가시키기도 하며, 사용될 블록의 적절한 수 및 계층 구조를 최적화해 줄 수 있는 코드 작업이 추가로 수행되어야 한다.

이러한 단점을 극복하는 한 가지 방법은 격자 세밀화를 블록 단위가 아닌 셀단위로 수행하는 것이다⁽⁵⁾. 셀단위로 적응 격자 세밀화를 수행하면 불필요한 격자점들이 세밀화 되는 것을 막을 수 있으며 이는 곧 계산 시간의 절감으로 이어진다.

그러나 셀단위로 적응 격자 세밀화가 이루어지

는 경우에는 세밀화된 격자점들 사이의 연결관계가 정렬화되어 있지 않기 때문에 이들간의 연결관계를 계층 데이터 구조를 통하여 따로 처리해 주어야 한다. 이는 비정렬 격자계 혹은 Cartesian 격자계의 격자간 연결 정보 처리와 동일한 형태라 할 수 있다.

2.3 Add-and-blank lines AMR algorithm

앞의 두 가지 적응 격자 기법은 유한 체적법(FVM)에 기반하여 제안된 방법인데 반해, 유한 차분법(FDM)에 기반하여 제안된 기법 중 본 연구에서 제안하고자 하는 방법과 일맥상통하는 기법이 제안된 바 있다⁶⁾.

이 기법의 핵심 아이디어는 블랭킹이라는 개념을 도입하였다는 것이다. 블랭킹의 개념은 중첩 격자 기법에서 주로 사용되는 것으로서 중첩된 격자들 중 일부의 격자계 내에 해석이 필요 없는 부분(고체 형상 등)이 존재하는 경우에 이를 해석 영역에서 제외시키는 방법이다. 즉, 격자 세밀화가 필요한 부분에는 라인을 추가하여 세밀한 격자를 생성하고, 세밀화가 필요 없는 부분은 블랭킹을 통해 계산영역에서 제외시키는 방법이다.

이 기법의 장점은 블록 단위로 세밀화 할 때 포함되는 불필요한 격자점들을 제거할 수 있어 계산 효율성이 좋으며, 앞서 설명한 셀단위 적응 격자 세밀화에서 수반되어야 하는 연결 관계 처리용 계층 데이터 구조가 필요 없다는 것이다. 다만, 블랭킹 과정에서 나타나는 행잉노드(hanging-node) 처리에 대한 복잡성이 있고, 유한 체적법으로 적용시에 대한 고려가 필요하다는 단점을 가진다.

3. 효율적인 셀단위 적응 격자 알고리즘

앞서 소개한 적응 격자 기법들의 특징을 통해 판단할 수 있는 것은 셀단위 세밀화를 통해 불필요한 격자점의 포함을 막고, 정렬 격자계의 인덱싱 정보를 이용함으로써 연결 관계 처리용 계층 데이터 구조를 대체할 수 있다면 계산 시간 및 메모리 면에서 효율성이 보다 향상될 수 있다는 점이다. 이에 본 연구에서는 중첩 격자 기법에서 사용되는 hole cell, fringe cell, normal cell의 개념을 도입하여 보다 효율적인 셀단위 적응 격자 알고리즘을 제안하고자 한다.

3.1 Grid description

일반적으로 적응 격자 알고리즘에서는 격자 세밀화를 수행할 격자점들을 특정 기준을 통해 먼저 선택하며, 이 과정을 플래깅(flagging) 작업이라고 한다. 예를 들어 Fig. 3(a)에서와 같이 기본 격자계에 놓인 한반도 내부 격자점들을 세밀화하고 싶다면 이에 대하여 Fig. 3(b)에서와 같이 플래깅을 수행한다.

다음으로 플래깅을 통해 선택된 격자점들에 대하여 어떠한 방법으로 적응 격자 세분화를 수행하느냐에 따라 2장에서 살펴본 바와 같이 여러 알고리즘으로 구분된다. 블록 단위로 격자 세분화를 수행하는 경우는 Fig. 4(a)와 같은 형태로 세분화가 진행되며, 셀단위 세분화는 Fig. 4(b)의 형태로 표현될 수 있다. 반면 본 연구에서 제안하고자 하는 효율적인 셀단위 적응 격자 기법에서는 Fig. 4(c)의 형태로 격자 세분화를 수행하게 된다. 즉, Fig. 3(b)에 표시된 한반도 내 격자점들을 모두 포함하는 직사각형 블록을 먼저 설정한 후, 블록 내부의 모든 격자점들에 대해서 세밀화를 수행한다. 다음으로 Fig. 3(b)에서 표시하였던 실제 세밀화가 필요한 부분은 normal cell로 표시하여 수치 플럭스 및 시간 적분 계산을 수행한다. Normal cell과 인접한 격자점들은 normal cell에 대한 경계로써 fringe cell로 설정하여 기본 격자계의 내삽을 통해 물성치를 설정하는 반면 플럭스 계산등의 수치 적분은 수행하지 않는다. 마지막으로 normal cell과 fringe cell을 제외한 부분은 hole cell로써 계산 영역에서 제외시킨다.

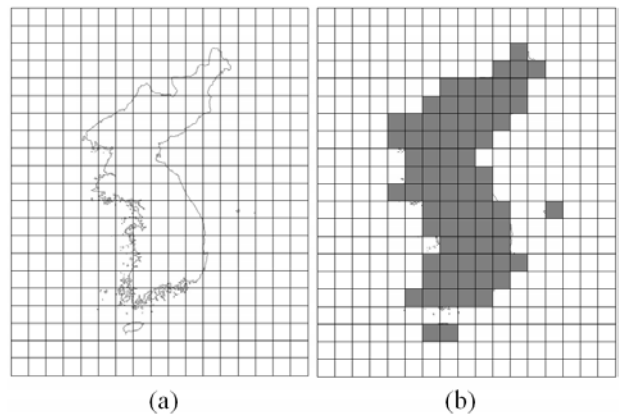


Fig. 3. An example of the area where higher resolution is required: (a) actual domain (the Korea Peninsula), (b) discretized domain (gray cells)

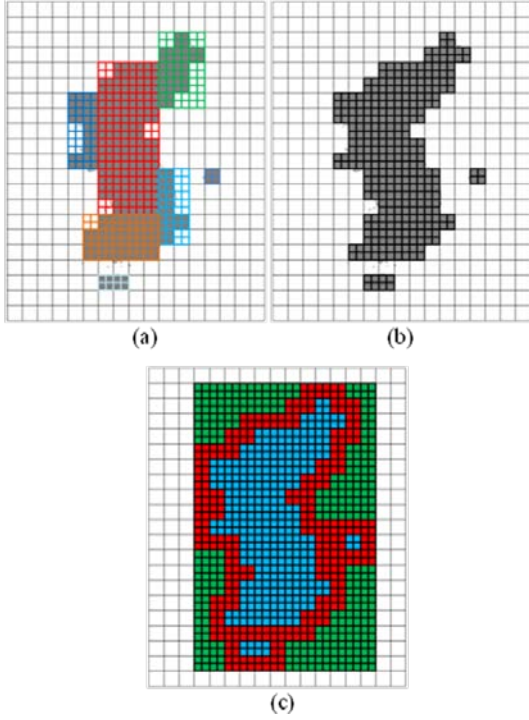


Fig. 4. Comparison of refined sub-domain for the Korean Peninsula: (a) block-by-block AMR method, (b) cell-by-cell AMR method, (c) present method (blue:normal cell, red:fringe cell, green: hole cell)

이러한 방법을 통해 불필요한 격자점의 포함을 막을 수 있으며, 또한 초기에 블록 형태로 설정한 내부의 세밀한 격자들 간에는 정렬 격자 인덱싱을 그대로 적용할 수 있으므로 연결 관계 처리용 계층 데이터 구조가 필요치 않다. 즉, normal cell, fringe cell 및 hole cell 을 판단할 추가적인 변수값 정의로 연결 관계 처리용 계층 데이터의 대체가 가능하여 매우 효율적이다.

3.2 Integration algorithm

세밀화된 적응 격자에 대해 수행할 수치 플럭스 및 시간 적분 방법은 블록 단위 기법에 사용된 방법⁽⁴⁾을 그대로 적용할 수 있으며, 이를 위해 먼저 보존형의 외재적 유한 체적 기법을 다음과 같은 형태로 표현한다.

$$u_{i,j}^{n+1} = u_{i,j}^n - \frac{\Delta t}{\Delta x} (F_{i+1/2,j} - F_{i-1/2,j}) - \frac{\Delta t}{\Delta y} (G_{i,j+1/2} - G_{i,j-1/2}) \quad \dots (3.1)$$

기본 격자점들 및 세밀화된 격자점들은 위의 식을

통해 각기 독립적으로 계산을 수행할 수 있다. 다만, 세밀화된 격자점들의 초기값을 설정할 때에는 내삽을 수행하여야 하며, 세밀화된 격자점들의 값을 이용하여 그 보다 큰 격자점의 값을 구할 때에는 다음 식과 같은 보존적 평균식을 이용한다.

$$u_{i,j}^{coarse} \leftarrow \frac{1}{\lambda_r^2} \sum_{p=0}^{\lambda_r-1} \sum_{q=0}^{\lambda_r-1} u_{k+p,m+q}^{fine} \quad \dots (3.2)$$

그리고, 세밀화된 격자계에 인접한 격자점들의 경우, 세밀화된 격자 경계에서 계산된 수치 플럭스의 합으로써 수치 플럭스를 구하게 된다(식 3.3).

$$u_{i,j}(t + \Delta t_c) = u_{i,j}(t) - \frac{\Delta t_c}{\Delta x} \left[F_{i+1/2,j}(t) - \frac{1}{\lambda_r^2} \sum_{p=0}^{\lambda_r-1} \sum_{q=0}^{\lambda_r-1} F_{k+1/2,m+p}(t + q\Delta t_f) \right] - \frac{\Delta t_c}{\Delta y} [G_{i,j+1/2}(t) - G_{i,j-1/2}(t)] \quad \dots (3.3)$$

세밀화된 격자점들과 이보다 큰 격자점들은 각기 독립적으로 계산이 수행되기 때문에 아래와 같이 식(3.4)를 적용 후, 식(3.5)를 계산함으로써 코드상에 구현하는 것이 일반적이다.

$$\delta F_{i+1/2,j} := -F_{i+1/2,j}^{coarse} \quad \dots (3.4)$$

$$\delta F_{i+1/2,j} := \delta F_{i+1/2,j} + \frac{1}{\lambda_r^2} \sum_{p=0}^{\lambda_r-1} F_{k+1/2,m+p}^{fine} \quad \dots (3.5)$$

4. 수치 실험 및 결과

본 연구에서 제안한 적응 격자 기법의 정확성 및 효율성을 수치 실험을 통해 확인해 보았으며, 적응 격자 기법을 사용하지 않은 경우와의 비교를 수행하였다. 공간 차분 기법으로는 RoeM 기법을 사용하였으며, 시간 차분 기법으로는 3 차 정확도 TVD Runge-Kutta 기법을 적용하여 비정상 유동을 계산하였다.

4.1 Isentropic vortex advection

이 수치 실험은 등엔트로피 와류가 비점성 자유류에 의하여 이동하는 문제로서, 수치 기법의 정

확도 및 격자 간격에 따라 와류 형상의 퍼짐 현상이 민감하게 나타나는 문제이다. 이러한 특성 때문에 수치 알고리즘의 정확성을 비교하는데 널리 사용되어진다.

초기 조건으로는 자유류 유동장에 등엔트로피 와류가 부과되는데, 자유류 및 등엔트로피 와류에 의한 변화치는 다음과 같다.

$$u_\infty = 1, \quad v_\infty = 1, \quad p_\infty = \rho_\infty = T_\infty = 1,$$

$$(\delta u, \delta v) = \frac{\beta}{2\pi} e^{(1-r^2)/2} (-\bar{y}, \bar{x}),$$

$$\delta T = -\frac{(\gamma-1)\beta^2}{8\gamma\pi} e^{1-r^2}.$$

여기서 와류 강도 $\beta=5$ 를 적용하였으며, \bar{x}, \bar{y} 는 각각 와류 중심으로부터의 x, y 좌표거리를 의미한다. 위의 식과 등엔트로피 관계를 이용하여 보존 변수의 엄밀해를 계산할 수 있다.

Fig. 5.는 와류 중심을 지나는 밀도 분포의 시간에 따른 변화를 보여준다. Fig. 5(a)에서 볼 수 있듯이 적응 격자를 사용하지 않은 경우, 시간에 따라 와류의 강도가 현저히 약해지는 반면, 적응 격자를 사용한 Fig.5(b)에서는 와류 강도를 유지하는 정도가 눈에 띄게 향상된 것을 확인할 수 있다.

Table 1.은 적응 격자를 사용하지 않은 경우 및 적응 격자의 상대적 크기비(λ_r)에 따른 에러의 크기와 계산 시간을 비교한 것이다. 우선, 적응 격자를 사용함으로써 전체적으로 에러의 크기가 줄어드는 것을 정량적으로 확인할 수 있으며, 적응 격자 크기비(λ_r)에 따른 에러의 크기도 확인할 수 있다. 흥미로운 사실은 λ_r 값이 커짐에 따라 에러의 값이 오히려 증가하기도 한다는 부분인데, 이는 적응 격자에 물성치를 부여하기 위하여 내삽을 하는 과정에서 격자 크기 차이가 커짐으로써 내삽의 정확도가 훼손되어 나타나는 현상으로 생각된다. 따라서 λ_r 값은 적정 크기를 넘어서지 않는 것이 계산의 효율성에 도움이 된다고 판단할 수 있다.

이러한 계산 효율성에 대한 부분을 보다 정량적으로 비교해 보기 위하여 Table 1.의 내용을 바탕으로 단위 계산 시간의 증가에 따른 에러의 감소량을 Table 2.에서 비교해 보았다. 즉, 단위 계산 시간의 증가에 대해 에러의 감소량이 클수록 기본 격자에 대해 추가된 격자가 효율적으로 사용되고

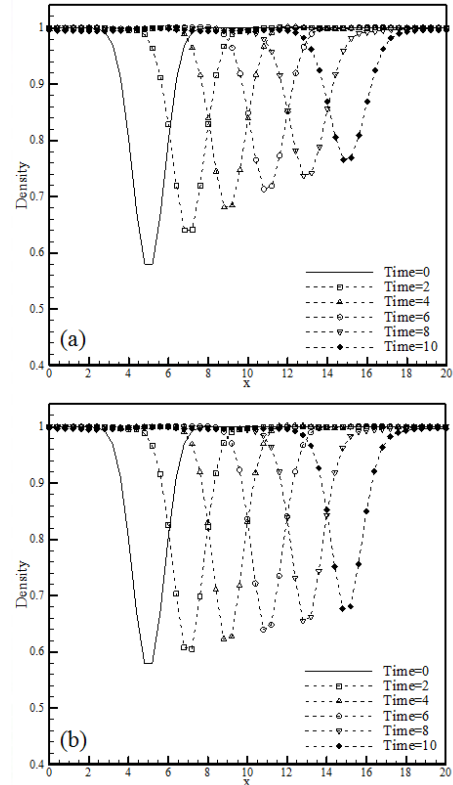


Fig. 5. Density distribution of the vortex along the center line at $t=0, 2, 4, 6, 8,$ and 10 : (a) 50×25 main grid, (b) 50×25 main grid with AMR ($\lambda_r=2$)

Table 1. Comparison of errors in isentropic vortex advection problem. Errors according to refinement ratio λ_r are compared.

size	AMR	L_1 error	L_2 error	L_∞ error	cost
$h=2/5$	-	5.67E-3	5.24E-4	2.11E-1	1
$h=2/5$	$r=2$	4.09E-3	3.26E-4	1.06E-1	1.79
$h=2/5$	$r=4$	3.63E-3	2.92E-4	1.15E-1	7.09
$h=2/5$	$r=8$	3.73E-3	3.05E-4	1.20E-1	48.3
$h=1/5$	-	1.67E-3	5.87E-5	4.94E-2	7.9

Table 2. Comparison of h -refinement efficiency for isentropic vortex advection

size	AMR	$\Delta L_1 / \Delta \text{cost}$	$\Delta L_2 / \Delta \text{cost}$	$\Delta L / \Delta \text{cost}$
$h=2/5$	-	-	-	-
$h=2/5$	$r=2$	-2.0E-3	-2.51E-4	-1.33E-1
$h=2/5$	$r=4$	-3.3E-4	-3.82E-5	-1.57E-2
$h=2/5$	$r=8$	-4.10E-5	-4.64E-6	-1.93E-3
$h=1/5$	-	-5.79E-4	-6.75E-5	-2.34E-2

있다고 판단할 수 있다. Table 2.에서 볼 수 있듯이 $\lambda_r=4$ 와 $\lambda_r=8$ 의 경우에는 단순히 전체 격자 크기를 절반으로 줄인 경우보다도 단위 계산량 증가당 에러의 감소가 적은 것을 확인할 수 있다. 반면, $\lambda_r=2$ 인 경우에는 전체 격자 크기를 절반으로 줄인 경우에 비해 3 배 이상 계산 효율성이 뛰어난 것을 확인할 수 있다.

4.2 Shock-vortex interaction (SVI)

다음으로 와류가 정상 충격파와 상호 작용하는 문제에 대하여 수치 실험을 수행하였다. 와류가 자유류에 의해 움직이면서 정상 충격파와 상호 작용하는 경우, 와류의 구조가 변화될 뿐 만 아니라 음향파와 충격파가 동시에 존재하는 복잡한 유동 구조가 나타난다. 계산 영역은 $[0, 2] \times [0, 1]$ 이며, 초기에 $x=0.5$ 위치에 정상 수직 충격파가 존재한다. 와류는 초기에 $(x_c, y_c) = (0.25, 0.5)$ 에 위치하다 자유류를 따라 이동하게 된다.

Fig. 6.은 계산된 결과의 $t=0.7$ 에서 수치 슐리엔 이미지를 보여준다. Fig. 6(a)는 적응 격자를 사용하지 않은 경우이며, Fig. 6(b)는 적응 격자를 사용한 결과로써 적응 격자를 사용하지 않은 경우에 비하여 복잡한 유동 구조의 세부 형태를 훨씬 더 잘 표현하고 있는 것을 볼 수 있다. 특히, 앞선 수치 실험에서 설명하였듯이 와류의 경우에는 격자 크기 및 이와 관련된 에러에 민감하게 형태가 나타나는 것이 일반적이며, 실제로 Fig. 6(b)에서도 와류의 형태가 두 개로 분리되는 세부 사항이 Fig.6(a)에 비해 월등히 잘 묘사되고 있음을 확인할 수 있다.

5. 결론

본 연구에서는 정렬 격자 기반의 효율적인 셀단위 적응 격자 알고리즘을 제안하였다. 제안된 셀단위 적응 격자 알고리즘은 중첩 격자 기법에 사용되는 hole cell, fringe cell, normal cell 의 개념을 도입함으로써 불필요한 격자점들의 세밀화를 막고, 동시에 정렬 격자 인덱싱 정보를 사용하여 추가적인 연결 관계 처리용 계층 데이터 구조 계산이 필요 없어 계산 시간 및 메모리 측면에서 효율적이라 판단된다. 수치 실험을 통하여 정확성 및 효율성을 확인해 보았으며, 적응 격자 크기비 설정 시에는 주의가 요구되는 것을 확인하였다.

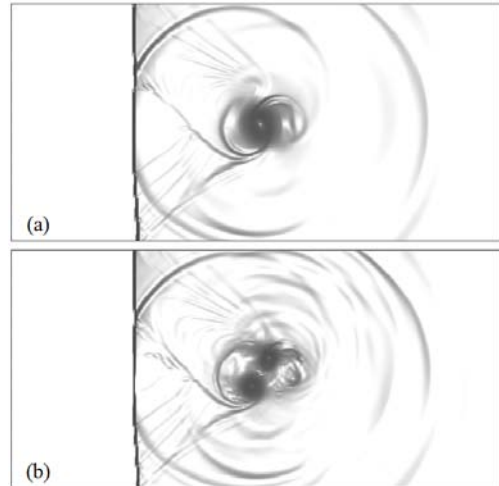


Fig. 6. Numerical Schlieren of shock-vortex interaction (SVI) at $t=0.7$: (a) 400×200 main grid, (b) 400×200 main grid with AMR ($\lambda_r=2$)

후 기

이 논문은 2011 년도 2 단계 두뇌한국 21 사업 및 교육과학기술부의 우주 기초 원천 기술 개발 사업 (NSL, 과제번호 2011-0029871), 그리고 한국연구재단의 지원(No. 2011-0027486)을 받아 수행된 연구이며 이에 감사 드립니다.

참고문헌

- (1) D. J. Mavripilis, Unstructured mesh generation and adaptivity, in von Karman Inst. for Fluid Dynamics, Lecture Series 1995.02 (AGARD Publication, 1995).
- (2) R. Hagmeier and J. Kok, Adaptive generation of structured grids, in von Karman Inst. for Fluid Dynamics, Lecture Series 1996.06 (AGARD Publication, 1996)
- (3) Marsha J. Berger and Joseph Olinger, Adaptive Mesh Refinement for Hyperbolic Partial Differential Equations, *J. Comput. Phys.* 53 (1984) 484-512
- (4) M. J. Berger and P. Colella, Local Adaptive Mesh Refinement for Shock Hydrodynamics, *J. Comput. Phys.* 82 (1989) 64-84
- (5) J.M. Morrell, P.K. Sweby and A. Barlow, A cell by cell anisotropic adaptive mesh ALE scheme for the numerical solution of the Euler equations, *J. Comput. Phys.* 226 (2007) 1152-1180
- (6) P. A. Durbin and G. Iaccarino, An Approach to Local Refinement of Structured Grids, *J. Comput. Phys.* 181 (2002), 639-653