

GPU를 이용한 반복적 물리 광학법의 가속화에 대한 연구

Acceleration of the Iterative Physical Optics Using Graphic Processing Unit

이 용 희 · 진 희 철 · 김 경 태

Yong-Hee Lee · Huicheol Chin · Kyung-Tae Kim

요 약

본 논문은 공동구조의 RCS(Radar Cross Section)을 계산하는 반복적 물리 광학법(Iterative Physical Optics: IPO)의 연산속도를 가속하는 기법들을 효과적으로 적용하는 방법을 제시한다. IPO는 기존에 공동 구조 내부에서 발생하는 다중 반사 효과 계산 시 기하 광학법(Geometric Optics: GO)를 사용하는 SBR(Shooting and Bouncing Rays)과는 달리 근거리 필드 식을 활용하기 때문에 정확도가 향상된 산란 계산이 가능하다. 하지만 PO(Physical Optics)에 비해 크게 느리며, 실질적인 사용을 위해서는 계산속도의 향상을 위한 기법이 필요하다. 이를 해결하기 위해 IPO에서 특징적으로 사용되는 반복적 부분을 GPU(Graphic Processing Unit)으로 계산하고, AIPO-CR(Adaptive Iterative Physical Optics-Change Rate)으로 반복횟수를 최적화하여 효과적으로 연산속도를 향상시킨다.

Abstract

This paper shows the acceleration of iterative physical optics(IPO) for radar cross section(RCS) by using two techniques effectively. For the analysis of the multiple reflection in the cavity, IPO uses the near field method, unlike shooting and bouncing rays method which uses the geometric optics(GO). However, it is still far slower than physical optics(PO) and it is needed to accelerate the speed of IPO for practical purpose. In order to address this problem, graphic processing unit(GPU) can be applied to reduce calculation time and adaptive iterative physical optics-change rate(AIPO-CR) method is also applicable effectively to optimize iteration for acceleration of calculation.

Key words: IPO, CUDA, AIPO-CR, RCS

I. 서 론

물체의 RCS(Radar Cross Section)을 계산하는 과정 중 공동구조에서 일어나는 전자기 산란의 해석은 중요한 부분을 차지한다. 특히 현대 전투기와 같이 낮은 RCS를 요구하는 상황에서 제트엔진의 흡입구 등의 공동구조가 전체 RCS에 미치는 영향을 분석하는 것은 매우 중요하다. 일반적으로 전자기 산란분석을 위하여 사용되는 대표

적인 기법은 PO(Physical Optics)로 입사파에 의해 유도된 표면 전류를 계산하고, 이를 토대로 방사되는 산란파를 계산하는 방식이다. 넓은 대형 구조물의 경우, 효과적으로 계산이 가능하나, 공동구조가 존재하게 되는 경우, PO의 계산은 다중반사로 인해 정확도가 감소한다.

정확도가 감소하는 문제는 맥스웰 방정식(Maxwell's equation)을 사용하는 MM(Moment Method)나 FDTD(Finite Different Time Domain) 등을 사용하여 해결이 가능하다.

포항공과대학교 전자공학과(Department of Electrical Engineering, Pohang University of Science and Technology)

· Manuscript received August 6, 2015 ; Revised November 11, 2015 ; Accepted November 13, 2015. (ID No. 20150806-054)

· Corresponding Author: Kyung-Tae Kim (e-mail: kkt@postech.ac.kr)

그럼에도 불구하고, 분석 가능 대상의 크기가 일반적으로 수 λ 내로 한정되며, 계산 속도가 느리고 큰 메모리공간을 요구하기 때문에 실제적으로 활용이 어렵다.

위 한계점들을 보완하기 위하여 SBR(Shooting and Bouncing Ray)을 사용할 수 있으며, PO의 경우보다 정확도가 향상된다^[1]. 하지만 기하 광학법(geometric optics)을 사용하여 다중 반사 효과를 계산하기 때문에, 이 역시 반사 횟수가 많을 경우 해석이 부정확하다.

SBR에서 제기된 문제점들을 해결하기 위해 IPO가 제안되었으며, 이 기법은 기하 광학법 대신 근거리 필드를 사용한다. 특히 다중 반사 효과 계산 시 발생하는 오차가 줄어들어 상기 언급된 MM이나 FDTD에서 계산된 결과와 유사하며, 분석가능 대상의 크기한계나 계산속도, 메모리 공간의 확보문제점들도 개선이 된다^{[2]-[4]}. 그럼에도 불구하고 여전히 계산속도는 기존 산란해석에 사용되던 PO에 비해 크게 느리며, 실제적으로 HRRP(High Resolution Range Profile)나 ISAR(Inverse Synthetic Aperture Radar) 영상 제작과 같이 주파수, 각도 영역에서 다량의 RCS 데이터가 필요한 경우에는 IPO를 사용하더라도 매우 많은 시간이 소요된다.

본 논문은 실질적인 공동구조의 RCS 계산을 위해 IPO에서 특징적으로 사용되는 반복적 부분에 GPU(Graphic Processing Unit)를 사용하여 병렬화 하고 GPU 환경에서 입사 각도별 반복횟수를 최적으로 정하는데 적합한 AIPO-CR(Adaptive Iterative Physical Optics-Change Rate)를 적용하여 메모리 사용량을 줄이며, 효과적으로 연산속도가 향상되는 것을 확인한다.

II. 본 론

2-1 반복적 물리 광학법(IPO)

물체의 RCS를 구하기 위한 공동구조의 전자기 산란 해석은 먼저 그림 1의 공동구조 입구(open)에 입사된 필드(\vec{E}^i, \vec{H}^i)에 등가원리(equivalent principle)을 사용하여 표면 등가 전류 $\vec{J}(\vec{r}')$, $\vec{M}(\vec{r}')$ 을 계산한다.

계산된 등가 전류에 의해 다시 자기장을 형성시켜 공동구조 내부 벽면에 유기되는 전류 값을 계산할 수 있다.

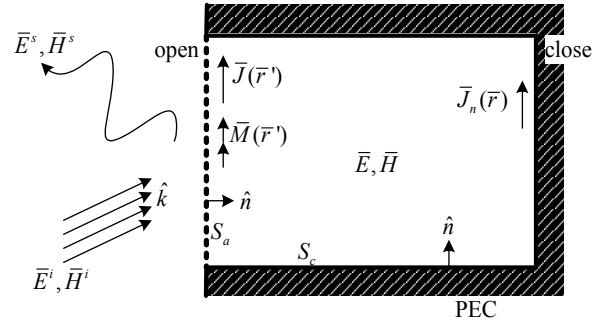


그림 1. 공동구조의 IPO 진행 모델

Fig. 1. IPO progress model of the cavity geometry.

$$\begin{aligned} \vec{H}(\vec{r}) \approx & \int_{S_a} \vec{J}(\vec{r}') \times \nabla \overline{G}_0(\vec{r} - \vec{r}') dS_a' \\ & + \frac{1}{jkZ_0} \int_{S_a} \vec{M}(\vec{r}') \times \nabla \overline{G}_0(\vec{r} - \vec{r}') dS_a' \end{aligned} \quad (1)$$

$$\nabla G(\vec{R}) = -\hat{R} \left(jk + \frac{1}{R} \right) \frac{e^{-jkR}}{4\pi R} \quad (2)$$

완전 도체를 가정하기 때문에 표면에는 위의 식 (1)과 같은 자계 적분 방정식을 이용하여 계산된 자기장이 유도되며, 이를 $\vec{J}(\vec{r}) = \hat{n} \times \vec{H}(\vec{r})$ 에 대입하면 공동구조 내부 벽면에 유기되는 전류량을 구할 수 있다. 앞서 자기장 계산에 사용된 식 (2)의 자유공간 그린함수(free space Green's function)은 기하 광학법을 대신해 근거리 필드 식을 사용하기 때문에 기존 SBR에 비해 정확도가 향상된다.

공동구조 내부 벽면에 유기된 전류의 값은 식 (3)의 자계적분 방정식에 표현되어 있으며, 좌변과 우변이 각각 n 차, $n-1$ 차의 전류를 표현하기 때문에 전류의 값이 수렴할 때까지 식 (3)을 반복한다.

$$\begin{aligned} \vec{J}_n(\vec{r}) = & 2\hat{n} \times \vec{H}(\vec{r}) - 2\hat{n} \times \oint_{S_c} \vec{J}_{n-1}(\vec{r}') \\ & \times \nabla G(\vec{r} - \vec{r}') dS_c' \end{aligned} \quad (3)$$

전류량이 수렴된 후에는 수렴된 전류에 의해 재 방사된 전파에 의해 공동 구조 입구에 유기된 등가 전류 $\vec{J}(\vec{r}')$, $\vec{M}(\vec{r}')$ 을 다시 계산하여 방사된 필드를 구하면 공동구조에 의한 전자기 산란 해석이 마무리 된다.

2-2 GPU를 활용한 가속화

GPU 프로세서는 CPU(Central Processing Unit)에 비해 프로세서의 개별속도는 느리지만, 다수의 프로세서를 활용하여 동시에 여러 개의 연산을 수행할 수 있기 때문에, 유사한 계산과정이 반복되는 상황에 적용하면 프로그램의 전체 속도를 향상시킬 수 있다.

IPO의 경우, 모델링을 구성하고 있는 각각의 삼각형 패치에 유도된 전류에 의해 다른 패치에 주는 영향을 모두 계산하기 때문에 공동 구조의 패치의 개수가 N 개일 경우 $O(N^2)$ 의 큰 계산량을 가지게 된다. 하지만 각각의 패치가 수행하는 계산 과정이 서로 동일하기 때문에 이를 다수의 GPU 프로세서에 할당시켜 병렬로 작업할 수 있게 되면 $O(N^2)$ 가 예상되더라도 GPU 프로세서의 개수만큼 이득(gain)을 가지게 된다.

CPU와 달리 GPU는 다양한 메모리 자원을 필요에 따라 사용할 수 있는 특징을 가진다. 레지스터, 공유 메모리, 글로벌 메모리, 상수 메모리가 있으며, 특성에 따라 알고리즘의 흐름에 적합한 메모리를 선택하여 사용하게 된다. 일반적으로 프로그램의 성능에 가장 큰 영향을 미칠 수 있는 글로벌 메모리는 전체 프로그램이 동작하는 동안 데이터를 보존할 수 있는 특징이 있으나, 프로세서가 접근하는 시간이 다른 메모리에 비해 오래 걸리며, 메모리공간 또한, 제한적이기 때문에 글로벌 메모리에 접근하는 횟수를 줄이는 것이나, 메모리 사용량 절약은 효과적인 프로그래밍의 요소가 될 수 있다. 글로벌 메모리에 관한 내용은 다음에 나올 2-3절에서 추가적으로 설명한다⁵⁾.

GPU를 사용하기 위해 프로그램의 시작부분에 kernel이라는 함수를 작성하며, GPU의 연산이 필요한 부분에서 kernel을 호출하게 된다. 그림 2는 본 논문에서 사용된 AIPO-CR이 적용된 IPO의 흐름도이며, 4번째 단계에서 AIPO-CR에 의해 반복 여부가 결정된다. 좌측에 주황색 상자로 표현된 것처럼 전체 IPO 과정을 하나의 kernel에 담게 되면 GPU의 한정된 메모리 자원들이 효율적으로 사용되지 못한 채 kernel이 끝날 때까지 낭비되는 경우가 발생할 수 있으므로, 효율적인 자원사용을 위해 계산 과정을 그림 2의 우측에 주황색 상자로 표현된 것처럼 3개의 kernel로 분할하여 작성하였다. 공동구조 입구에서 등가전류를 계산하는 전 처리 단계, 공동구조 내부 벽면의 전류를 수립시키는 단계, 수립된 전류를 재 방사시키는 후

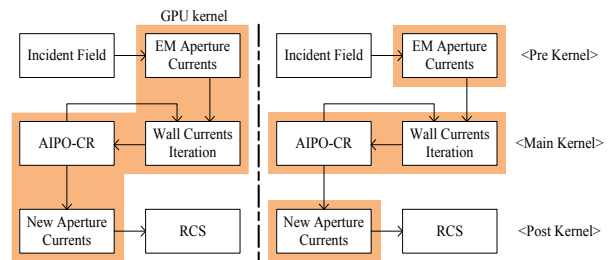


그림 2. GPU가 사용된 IPO의 흐름도 - 전/후
Fig. 2. Diagram of IPO using GPU - before/after.

표 1. Main kernel 알고리즘

Table 1. Main kernel algorithm.

Main_Kernel Algorithm
<p>Input: I, current data for patches; G, geometry data for patches; N, the number of patches;</p> <p>Output: none, an (intermediate) I data stored in I; $thread \leftarrow blockIdx.x * blockDim.x + threadIdx.x$; //Start global memory read early from current of patches. $initial_current \leftarrow I[thread]$;</p> <p>for $patch_number \leftarrow 0$ to N do $induced_current \leftarrow I[patch_number]$; if $patch_number \neq thread$ then $induced_current \leftarrow induced_current$ $+ near_field_green_function(induced_current, G)$; end if end for $I[thread] \leftarrow initial_current + induced_current$; return 0;</p>

처리 단계로 구분하여 GPU를 이용한 기법을 효율적으로 적용시킨다²⁾.

표 1은 그림 2에서 표현된 main kernel의 알고리즘을 나타낸 pseudo code이다. 그림 2의 우측에서 3개의 kernel로 나누어져 있지만, 3개 kernel 모두 핵심 알고리즘으로 근거리 필드 식을 사용하기 때문에 알고리즘이 유사하여 main kernel로 대표하여 나타내었다. 3개 kernel의 주요 차이는 근거리 필드식의 source와 observer가 매 호출 때마다 변경되는 것이다.

이 논문에서는 GPU를 사용하기 위하여 nVIDIA사에서 2007년에 고급언어 기반으로 개발된 CUDA 플랫폼을 활용하였다.

2.3 AIPO-CR을 이용한 반복횟수의 자동 계산

IPO 기법은 여러 각도에서 적용될 경우 수렴하는데 필요한 반복법의 횟수는 각도별로 상이하다. 그러므로 반복법의 횟수가 일관되게 적용된다면 일부 각도에서 반복의 횟수가 부족하거나, 과도할 수 있게 되며, 이는 각각 정확도와 속도의 하락을 가져오는 단점이 된다. 이를 보완하기 위하여 기존에 CPU를 활용한 IPO에서는 수렴 값을 찾는데 광선추적법(ray tracing)과 전류량의 차이를 비교하는 방법, 그리고 전류량의 에너지 차이를 이용하는 알고리즘들이 적용되었다^{[2],[6],[7]}.

광선추적법은 공동구조 입구에 일정한 간격의 광선 다발을 입사시켜 밖으로 방사될 때까지 각 광선이 내부에서 반사되는 횟수를 종합하여 가장 높은 비율을 가지는 횟수를 선택하는 방법이다. 직관적이지만 광선을 추적하는데 많은 계산비용과 알고리즘의 추가가 필요하다. 또한, 입사파의 편파를 고려하지 않기 때문에 다른 방법에 비해 효율적이지 않다^[2].

전류량의 차이(difference of current)를 계산하여 변화량을 비교하는 방법은 $\|J_n - J_{n-1}\| / \|J_0\|$ 와 같이 상대오차(relative error)를 구하는 식을 각각의 패치에 적용하여 변화율의 총합을 추적하는 것이며, 도출된 값이 정해진 기준 이하로 내려가면 반복시행이 중단되게 된다. 이 방식은 각각의 패치가 가지고 있는 전류성분을 반복 전후 (J_n, J_{n-1})로 그래픽 카드의 글로벌 메모리에 저장해야 함을 의미한다. 또한, 식 (3)의 계산을 위해 분모의 J_0 값 또한, 저장되어야 하며, 병목현상을 일으킬 수 있는 글로벌 메모리의 읽기/쓰기의 지연을 방지하기 위해 $J_n - J_{n-1}$ 도 따로 저장되어야 하기 때문에 전류량의 차이를 계산하는 간단한 연산에 대부분의 글로벌 메모리가 사용되는 것을 알 수 있다. 연산이 간단하다는 점을 고려하면 CPU를 사용한 IPO의 계산에서 효과적으로 반복횟수를 결정하는데 사용될 수 있으나, GPU 연산에서는 글로벌 메모리 접근 빈도수나 매우 제한적인 메모리공간의 낭비라는 관점에서 개선의 노력이 필요하다^{[5],[6]}.

위 상황을 개선하기 위해 사용한 AIPO-CR은 기존 CPU 기반에서 반복횟수를 결정하기 위해 사용하였던 것으로, 전류량 변화율의 총합을 전류량의 전체 에너지 합

으로 바꾸어 사용하는 알고리즘이다^[7].

n 번째에 계산된 전류의 전체 에너지는 식 (4)와 같으며, 식 (5)와 같이 $n-1$ 번째에 계산된 전류의 전체 에너지와 비교하여 반복법의 중단점을 결정하게 된다. 본 논문은 식 (5)의 $CR(n)$ 이 1%보다 작거나 같으면 식 (3)의 반복을 중단하도록 하였으며, 이 부분을 사용자가 조절하여 속도와 정확도라는 상반되는 두 측면 사이에서 원하는 정도를 결정할 수 있게 해준다.

$$E(\overline{J}_n) \equiv \sum_i^{all\ facets} \|\overline{J}_n(i)\| \quad (4)$$

$$CR(n) \equiv \left| \frac{E(\overline{J}_n) - E(\overline{J}_{n-1})}{E(\overline{J}_{n-1})} \right| \times 100\% \quad (5)$$

전체에너지의 합으로 변환된 값의 차이를 비교하여 반복횟수를 정할 수 있기 때문에 $E(\overline{J}_{n-1})$ 와 $E(\overline{J}_n)$ 실수 2개만 필요한 것을 알 수 있으며, 이 값은 각 반복법이 시행되는 전후로 계산하여 저장하면 된다. J_{n-1} 과 J_n 중 반복 전에 해당하는 J_{n-1} 은 반복횟수를 정하는데 더 이상 사용되지 않기 때문에, 표 1을 따라 반복법을 시행한 후 계산된 J_n 을 J_{n-1} 에 저장하면 메모리 공간이 절약된다. 하지만 이 경우 하나의 메모리공간에 반복 전후의 값을 연속하여 입력하게 되면, 글로벌메모리의 읽기/쓰기와 관련된 지연현상이 발생할 수 있는데 이는 표 1에서 확인할 수 있는 ‘initial_current ← [thread];’ 명령이 글로벌 메모리의 값을 초반에 미리 읽어 들여 그래픽카드의 레지스터에 저장하는 기법을 통해 피할 수 있다^[5]. 결과적으로 AIPO-CR은 J_n (혹은 J_{n-1})과 J_0 만 필요하게 되어, 전류량의 차이를 사용한 방법에 비해 사용 메모리가 50% 감소하는 것을 예상할 수 있다.

앞서 제시된 여러 방법들을 통해 계산한 반복횟수가 상이할 수 있으나, 기법들이 제안된 논문들을 통해 충분히 수렴하는 지점을 자동으로 계산해 주는 것을 확인하였으므로 결과의 정확도는 큰 차이가 없는 것을 예상할 수 있다^{[6]-[8]}. 그러므로 계산속도와 메모리공간을 절약하는 장점이 예상되는 AIPO-CR은 GPU를 사용하는 IPO의 반복횟수를 결정하는 알고리즘으로 가장 적합하다고 판단하였다.

III. 시뮬레이션 결과

3-1 시뮬레이션 조건

본 논문에 사용된 컴퓨터의 CPU는 Intel(R) Core(TM) i7-2600k CPU @ 3.40 GHz, RAM은 DDR3 6400 16.0 GB, GPU는 GeForce GTX 770의 사양을 가진다.

기법 간의 적용 효과를 확인하기 위하여 IPO만 단독으로 동작하였을 경우와 GPU가 적용된 경우, 그리고 GPU와 AIPO-CR이 동시에 적용되었을 경우를 가정한다. 분석 대상의 모델은 그림 3과 같으며 각각 θ 와 ϕ 편파로 분석을 시행한다. 그림 3과 같이 공동구조 입구에 수직한 방향에서 50도가 될 때까지 입사파방향 \hat{k} 벡터를 1도씩 변화시켜 전자기 산란을 IPO로 해석한다.

AIPO-CR이 적용되지 않을 경우, 수렴횟수를 사용자가 결정하기 때문에 수렴지점으로 판단되는 수준은 FEKO의 결과와 대조하여 결정하였으며, 짧은 모델의 경우 수렴횟

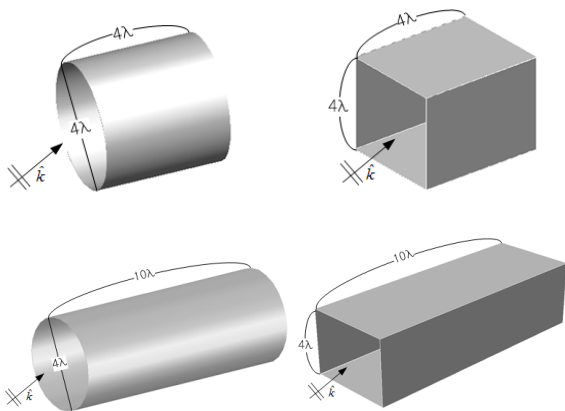


그림 3. IPO 대상 모델
Fig. 3. IPO target model.

수를 5회, 긴 모델의 경우 10회로 결정하였다^[8].

3-2에서는 기법 간의 결과를 FEKO와 대조하여 정확도를 확인한 후 가속화 효과를 정량화 하여 비교하며, 또한 AIPO-CR의 사용으로 인한 글로벌 메모리 사용량 감소를 확인한다.

3-2 기법에 따른 시뮬레이션 결과 비교

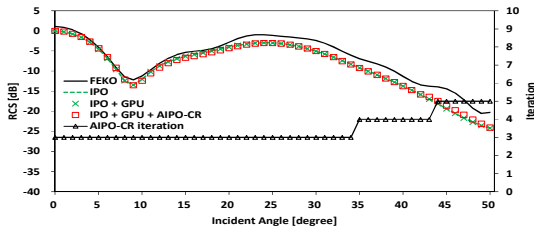
표 2는 기법 간의 결과들과 대조군으로 사용된 FEKO의 결과를 비교하여 RMSE(Root Mean Square Error)를 구한 것이다. RMSE는 기준값 대비 추정값이 가지는 차이에 대한 측도이며 $\sqrt{\sum_{n=1}^N (x_n - x'_n)^2 / N}$ 을 통해 구할 수 있다. 여기서 x 는 각 기법으로 계산한 추정값, x' 은 FEKO로 계산한 기준값 그리고 N 은 계산각도의 수를 나타낸다. 평균 RMSE는 2.77 dB이며, 정규화된 RMSE는 약 10% 내외가 되는 것을 확인할 수 있다. AIPO-CR 적용 유무에 따른 RMSE의 차이는 0.35 dB 내외로 수렴 정도가 유사함을 나타내며, 이는 또한, 그림 4에서 각 기법 간의 결과 그래프로 확인할 수 있다. 다만 이 경우 RMSE가 다소 증가하는 방향으로 나타나는 것을 확인할 수 있는데, 비교대상이 MOM을 사용한 FEKO 결과이기 때문에 IPO와 수렴결과가 부분적으로 차이가 날 수밖에 없는 것을 고려한다면 AIPO-CR에 의한 결과는 앞선 0.35 dB의 수치를 고려할 때 충분히 수렴하여 정확도가 유지되는 것으로 판단된다. GPU의 사용 유무는 기존 IPO 알고리즘을 단순히 병렬화 시키는 것이기 때문에, 결과의 정확도에 차이를 주지 않는 것을 표 2과 그림 4를 통해 확인할 수 있다.

표 3에 따르면 GPU가 적용되었을 때 짧은 모델의 경우, 평균 약 200배, 긴 모델의 경우 평균 약 800배 빨라지는 것을 확인하였으며, 이는 IPO에 GPU가 개입하는 시간이

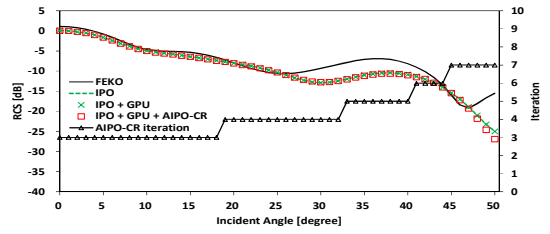
표 2. IPO에 적용된 기법들의 차이에 따른 결과의 RMSE(단위: dB)

Table 2. Computing RMSE of the various techniques applied to IPO(unit: dB).

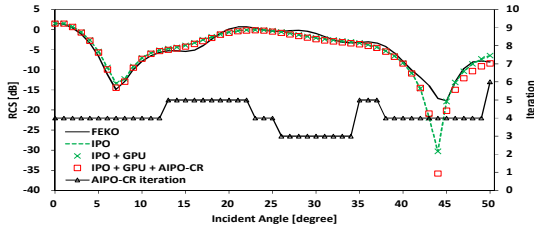
	Short cylindrical		Short rectangular		Long cylindrical		Long rectangular	
	θ -pol.	ϕ -pol.	θ -pol.	ϕ -pol.	θ -pol.	ϕ -pol.	θ -pol.	ϕ -pol.
IPO	2.32	2.58	2.36	2.27	2.64	3.37	1.96	3.36
IPO + GPU	2.32	2.58	2.36	2.27	2.64	3.37	1.96	3.36
IPO + GPU + AIPO-CR	2.19	2.84	3.02	2.30	2.86	4.17	2.43	3.63



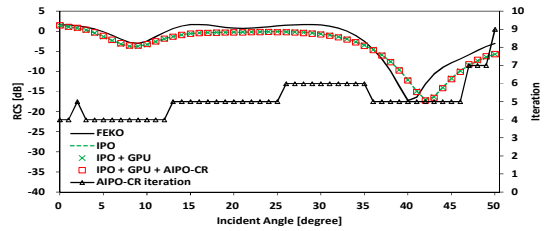
(a) 짧은 원형 공동구조(θ -pol.)
(a) Open-ended short cylindrical cavity (θ -pol.)



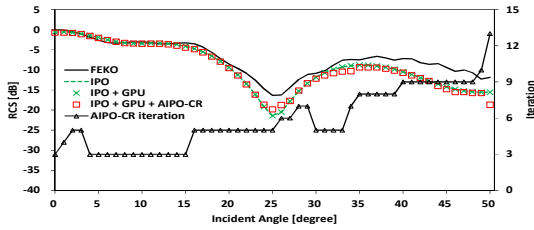
(b) 짧은 원형 공동구조(ϕ -pol.)
(b) Open-ended short cylindrical cavity (ϕ -pol.)



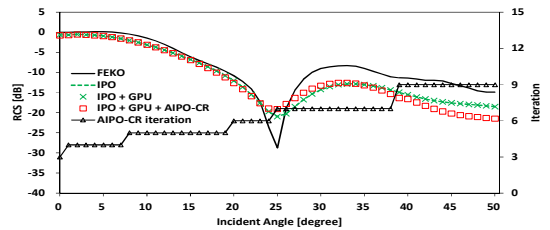
(c) 짧은 사각형 공동구조(θ -pol.)
(c) Short rectangular cavity (θ -pol.)



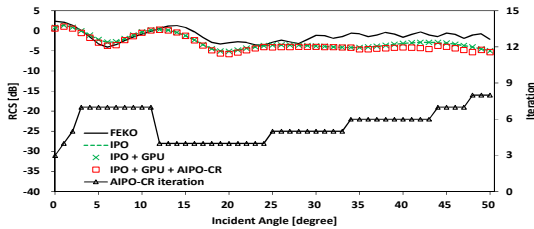
(d) 짧은 사각형 공동구조(ϕ -pol.)
(d) Short rectangular cavity (ϕ -pol.)



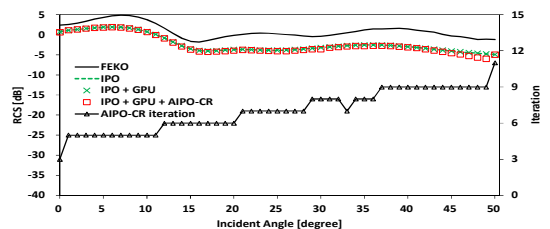
(e) 긴 원형 공동구조(θ -pol.)
(e) Open-ended long cylindrical cavity (θ -pol.)



(f) 긴 원형 공동구조(ϕ -pol.)
(f) Open-ended long cylindrical cavity (ϕ -pol.)



(g) 긴 사각형 공동구조(θ -pol.)
(g) Open-ended long rectangular cavity (θ -pol.)



(h) 긴 사각형 공동구조(ϕ -pol.)
(h) Open-ended long rectangular cavity (ϕ -pol.)

그림 4. IPO에 적용된 기법에 따른 결과그래프

Fig. 4. Result graph of the various techniques applied to IPO.

많이 늘어날수록 속도 효율이 좋아지는 것을 보여준다. 또한, 표 3에서 GPU와 AIPO-CR이 동시에 적용된 것을 보았을 때, 단순히 GPU만 적용한 상황보다 향상된 성능을 보여주는 것을 확인할 수 있으며, 이는 그림 4에서 표

현된 각도별 반복횟수의 차이에서도 확인이 가능하다. 모델별로 가해진 반복횟수와 대조해 볼 때, AIPO-CR을 사용한 경우 각도별 반복횟수가 감소하여 전체 동작시간이 감소하는 것을 확인할 수 있다. 다만 표 3의 short rectan-

표 3. IPO에 적용된 기법들의 차이에 따른 계산 성능(단위: 초)

Table 3. Computing performance of the various techniques applied to IPO(unit : second)

	Short cylindrical		Short rectangular		Long cylindrical		Long rectangular	
	θ -pol.	ϕ -pol.	θ -pol.	ϕ -pol.	θ -pol.	ϕ -pol.	θ -pol.	ϕ -pol.
IPO	107.58	127.56	472.66	472.89	2,478.79	2,480.10	4,366.81	4,369.82
IPO + GPU	0.86	0.83	1.54	1.50	4.25	4.23	5.75	5.77
IPO + GPU + AIPO-CR	0.72	0.81	1.28	1.56	2.58	2.82	3.47	3.09

표 4. 글로벌 메모리 사용량(단위: kByte)

Table 4. The usage of global memory(unit: kByte).

	Difference of current	AIPO-CR
Short cylindrical	41.0	20.4
Short rectangular	80.8	40.4
Long cylindrical	125.2	62.6
Long rectangular	165.1	82.6

gular의 ϕ -pol.에서 시간이 증가하는 것을 확인할 수 있는데, 이는 모델별, 상황별로 다를 수밖에 없는 반복횟수의 차이를 임의로 결정하기 어려움을 반증해 주는 것으로 그림 4의 (d)에서 AIPO-CR을 통하여 구한 반복횟수가 짧은 모델에 기본적으로 정한 5회의 반복횟수를 상회하는 것으로 확인할 수 있다.

표 4에서는 AIPO-CR이 적용된 경우, 전류량의 차이를 사용해 반복횟수를 정하는 경우를 가정한 알고리즘에 비해 사용된 글로벌 메모리의 사용량이 약 50 %로 감소한 것을 확인할 수 있다.

IV. 결 론

본 논문에서는 기존에 사용되던 IPO의 연산속도가 실질적으로 사용하기에는 느리다는 단점을 보완하기 위해 GPU와 AIPO-CR의 기법을 적용하였다. CPU의 적용을 통해 반복적으로 계산되는 부분을 병렬화 하고, AIPO-CR을 적용하여 전파의 입사 각도별 상이한 반복횟수를 수렴지점에 최적화 시켜 불필요한 계산을 막아 연산속도를 모델에 따라 약 200배에서 800배 가량 개선하였다. 또한, GPU가 활용된 IPO상에서 AIPO-CR을 통하여 글로벌 메모리의 사용량을 줄이는 것과 동시에 충분한 연산속도의

개선과 정확도가 유지되는 것을 확인하였다.

References

- [1] Hao Ling, Ri-Chee Chou, and Shung-Wu Lee, "Shooting and bouncing rays: calculating the RCS of an arbitrarily shaped cavity", *IEEE Transactions on Antennas and Propagation*, vol. 37. no. 2, Feb. 1989.
- [2] Fernando Obelleiro-Basteiro, Jose Luis Rodriguez, and Robert J. Burkholder, "An iterative physical optics approach for analyzing the electromagnetic scattering by large open-ended cavities", *IEEE Transactions on Antennas and Propagation*, vol. 43. no. 4, apr. 1995.
- [3] 임호, 최승호, 명로훈, "제트 엔진 산란 해석을 위한 하이브리드 IPO-MoM 기법", *한국전자과학회논문지*, 20(8), pp. 791-796, 2009년 8월.
- [4] 고일석, 박창현, 선웅, "IPO(Iterative PO)를 이용한 Ka 대역 Von Karman Radome 방사 특성 해석", *한국전자과학회논문지*, 22(12), pp. 1148-1154, 2011년 12월.
- [5] David B. Kirk, Wen-mei W. Hwu, *Programming Massively Parallel Processors: A Hands-on Approach*, Elsevier, 2010.
- [6] Robert J. Burkholder, Tomas Lundin, "Forward-backward iterative physical optics algorithm for computing the RCS of open-ended cavities", *IEEE Transactions on Antennas and Propagation*, vol. 53. no. 2, 793-799, 2005.
- [7] S. H. Choi, D. W. Soe, and N. H. Myung, "Scattering analysis of open-ended cavity with inner object", *Journal of Electromagnetic Waves and Applications*, vol. 21. no. 12, 1689-1702, 2007.
- [8] H. Chin, J. -H. Yeom, H. -T. Kim, and K. -T. Kim, "Im-

provement of iterative physical optics using previous information to guide initial guess", *Journal of Electro-*

magnetic Waves and Applications, vol. 124, 473-486, 2012.

이 용 희



2009년 2월: 포항공과대학교 전자공학과 (공학사)
2009년 3월~현재: 포항공과대학교 전자전기공학과 석·박사 통합과정
[주 관심분야] 전자파 산란 해석, RCS

김 경 태



1994년 2월: 포항공과대학교 전자전기공학과 (공학사)
1996년 2월: 포항공과대학교 전자전기공학과 (공학석사)
1999년 2월: 포항공과대학교 전자전기공학과 (공학박사)
2002년 3월~2011년 2월: 영남대학교 전자공학과 교수

2011년 3월~현재: 포항공과대학교 전자전기공학과 교수
2012년 9월~현재: 레이더/IR 표적식별 특화연구실 실장
[주 관심분야] 레이더 표적인식, 레이더 영상, 레이더 신호처리, 패턴인식, 전자기수치해석 및 RCS 추정

진 희 철



2008년 2월: 부산대학교 전자전기공학과 (공학사)
2014년 2월: 포항공과대학교 전자공학과 (공학박사)
2014년 3월~2014년 11월: (주)만도
2015년 1월~현재: 국방과학기술품질원
[주 관심분야] 전자파 산란 해석, 전자기

필스