

MLFMM 방법의 알고리즘

고 일 석 · 이 현 수

인하대학교 전자공학과

I. 서 론

레이다 산란 단면적(RCS: Radar Cross Section)은 전자파를 송·수신하여 미확인 물체를 식별하는 레이다 기술의 중요한 파라미터 중 하나이다. 미확인 물체(산란체)에 의한 산란파 전력[W]은 송신된 전파의 전력밀도[W/m²]와 RCS[m²]의 곱으로 나타낼 수 있다. RCS는 입사각, 관측각, 그리고 주파수의 함수이므로^[1,2], 산란파 또한 이런 파라미터의 함수이다. [그림 1]은 전자파가 레이다에서 송신되어 산란체를 거쳐 다시 수신되는 과정을 나타낸 모식도이며, 3 GHz 주파수에서 B-26의 RCS 예를 함께 나타내었다^[1].

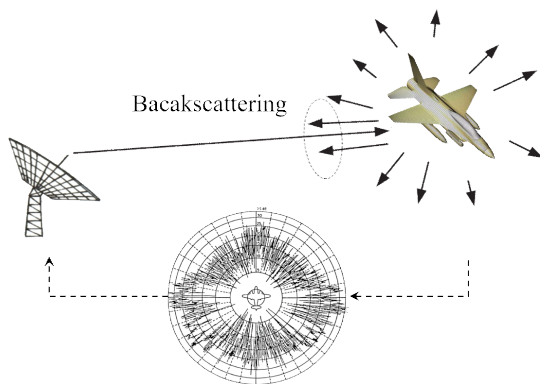
구, 원기둥, 정육면체 등 전형적 도형들에 대한 산란파는 해석적 해로 표현할 수 있다^[2]. 그러나 임의의 형상을 갖는 물체에 관한 산란파는 보통, 컴퓨터를 사용하여 수치해석방법으로 계산한다. 컴퓨터를 사용하여 산란체의 산란파를 계산하는 첫 번째 과정은 산란체 형상을 차분화(mesh)하는 과정이다. 산란체의 물리적 크기가 커지거나 주파수가 높아수록 메시(mesh)의 개수도 기하급수적으로 늘어난다. 메시의 개수가 늘어나면, 산란파를 구하기 위한 연산량과 연산시간

도 같이 증가한다. 이들을 줄이기 위하여 물리광학법(PO: Physical Optics)과 기하광학법(GO: Geometrical Optics) 등의 고주파 근사법이 사용되기도 하며, 이들의 연산량과 연산시간은 $O(N)$ 으로 알려져 있다^[2]. 여기서 N 은 메시의 개수이다.

위 방법들은 산란체의 전기적 길이(electrical length)가 커질수록 정확한 결과를 보인다. 일반적으로 다중반사파 또는 회절파 등을 정확히 고려하는 것은 매우 어렵고, 컴퓨터 자원도 많이 필요하다. 다중반사파나 회절파 등을 효율적으로 고려하기 위해 PO나 GO 방법들이 SBR(Shooting and Bouncing Rays), IPO(Iterative Physical Optics)^[3,4], UTD(Uniform Theory of Diffraction), 또는 PTD(Physical Theory of Diffraction)방법^{[5]~[8]}들로 확장되었다. 그러나 이런 방법들을 사용하는데 추가적인 연산량이 필요하고, 저 RCS 물체에 사용하기에는 정확도에 한계가 있다.

컴퓨터를 사용하여 정확한 수치해를 구하는 대표적인 방법으로 FEM(Finite Element Method)과 MoM(Method of Moments) 등이 있다. FEM과 MoM은 각각 산란해석 문제의 미분방정식과 적분방정식의 해를 구하는 방법이다. 두 방법들 모두 full-wave simulation이라 불린다. 하지만 이들의 연산량과 메모리 사용량은 $O(N^2)$ 으로 매우 높기 때문에^[9,10], 산란체의 전기적 크기가 커지면 사용하기 어렵다.

일반적으로 산란해석에서는 MoM 방법을 선호한다. MoM 방법을 대단위 물체에 적용하기 위한 연구는 오랫동안 수행되어 왔다. 1985년 V. Rokhlin에 의해 MoM 방법의 연산량을 $O(N^{1.5})$ 로 줄이는 방법이 제안되었고, 이 방법은 현재 FMM(Fast Multipole Method) 방법으로 알려져 있다^[11]. FMM 방법은 MoM 행렬 방정식의 해를 비선형 반복법(nonlinear iterative method)으로 구하고, 이때 필요한 벡터-행렬 곱의 연산량을 그룹 간 상호작용을 이용하여 $O(N^2)$ 에서 $O(N^{1.5})$



[그림 1] 산란파가 레이다에 도달하는 과정을 나타낸 모식도

이 논문은 2017년도 정부(과학기술정보통신부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임(No.2016-0-00130, RF 설계 및 EM 해석을 위한 클라우드 기반 SW 플랫폼 개발)

로 줄이는데 성공했다^{[12],[13]}. 이후 FMM 방법을 계층적으로 (hierarchically) 적용하여 벡터-행렬 곱의 연산량을 $O(N \log N)$ 까지 낮추었다. 이 방법을 MLFMM(Multi-Level Fast Multipole Method) 방법으로 불린다^{[14],[15]}. MLFMM은 연산량 외 필요메모리 요구량도 $O(N \log N)$ 수준으로 낮아 대규모 산란체에 적용 가능하여, 현재 산란 문제를 푸는 표준방법으로 FEKO, HFSS, VIRAF, CST 등 다양한 상용 소프트웨어들에 구현되어 있다.

본 논문에서는 기본 MLFMM 알고리즘을 간략히 소개한다. MLFMM 방법은 MoM 방법에 기반하여, 2-1장에서 MoM 방법의 과정을 간단히 살펴본 후에 MLFMM 방법의 차이점에 관하여 설명하겠다. 2-2장부터 2-6장까지는 MLFMM 방법에 사용되는 알고리즘들을 순차적으로 서술하겠다. 수식은 $\exp(j\omega t)$ 와 완전도체(PEC: Perfectly Electric Conductor) 매질을 가정하여 기술하겠다.

II. 본문: MLFMM 방법의 알고리즘

2-1 MoM vs MLFMM

PEC 산란체 표면에 경계조건을 적용하면, 전계에 관한 적분방정식(EFIE: Electric Field Integral Equation) (1)과 자계에 관한 적분방정식(MFIE: Magnetic Field Integral Equation) (2)를 얻을 수 있다.

$$\hat{n} \times \vec{E}^i = jk_0 Z_0 \hat{n} \times \iint_{S'} \left[\vec{J} + \frac{\nabla' \nabla' \cdot \vec{J}}{k_0^2} \right] \frac{e^{-jk_0 |\vec{r} - \vec{r}'|}}{4\pi |\vec{r} - \vec{r}'|} d\vec{r}' \quad (1)$$

$$\hat{n} \times \vec{H}^i = \frac{\vec{J}}{2} - \hat{n} \times P.V. \iint_{S' - \delta S} \vec{J} \times \nabla' \frac{e^{-jk_0 |\vec{r} - \vec{r}'|}}{4\pi |\vec{r} - \vec{r}'|} d\vec{r}' \quad (2)$$

여기서 \hat{n} 은 산란체 표면에서 밖을 향하는 법선벡터이고, \vec{E}^i 와 \vec{H}^i 은 입사파의 전계와 자계이고, k_0 는 자유공간의 파수(wave number)이고, Z_0 는 자유공간의 특성 임피던스이다. \vec{r} 과 \vec{r}' 는 각각 관측점과 소스의 위치벡터이고, P.V.는 적분의 principal value를 나타낸다.

입사파는 기지의(known) 값이고, 이에 의해 유도되는 산란체 표면전류 \vec{J} 는 미지의(unknown) 값이다. 일단 \vec{J} 를 구하면, 산란파는 radiation integral을 통하여 쉽게 얻을 수 있

다^{[2],[10]}. 표면전류 \vec{J} 를 모사하기 위하여, 산란체의 메시마다 적절한 기저함수를 부여하고, \vec{J} 를 이들의 선형결합으로 다음과 같이 나타낸다.

$$\vec{J} = \sum_{n=1}^N a_n \vec{f}_n \quad (3)$$

여기서 a_n 은 미지의 계수이고, \vec{f}_n 은 사용자가 선택한 기저함수이며, 주로 RWG 기저함수가 사용된다^[16]. 식 (3)을 식 (1)에 대입한 후, 계수 a_n 에 관하여 정리하면 미지수 N 개에 관한 식 N 개를 얻을 수 있다. 하지만 미지수 N 개의 해를 구하기 위해서는, 식 N 개가 필요하다. 만약 적분방정식의 관측점을 메시마다 적용하여 N 개의 식을 얻으면 point-matching 방법이라 부른다.

Point-matching 방법은 해당 관측점에서는 정확하나, 산란체 전체 영역에 걸쳐서는 부정확하기 때문에^{[2],[9]}, 적절한 테스트함수를 선택하고, 이를 식 (1)의 양변에 곱한 후 적분하여 weak-form을 만들어준다. Point-matching 방법은 테스트함수로 델타함수를 사용한 경우에 해당하고, 특히 기저함수와 동일한 테스트함수를 사용한 경우를 Galerkin 방법이라 불리고, 식 (1)과 (2)를 차별화하는 주된 방법이다. 이러한 일련의 과정을 거치면, 식 (4)와 같은 행렬식을 얻을 수 있다.

$$[V]_{N \times 1} = [Z]_{N \times N} [I]_{N \times 1} \quad (4)$$

$$V_m = \iint_{S'} \vec{f}_m \cdot (\hat{n} \times \vec{E}^i) d\vec{r}' \quad (5)$$

$$Z_{mn} = jk_0 Z_0 \iint_{S'} \vec{f}_m \cdot \left(\hat{n} \times \iint_{S'} \left[\vec{f}_n + \frac{\nabla' \nabla' \cdot \vec{f}_n}{k_0^2} \right] G d\vec{r}' \right) d\vec{r}' \quad (6)$$

여기서 V 는 N 차 열벡터이고, m 행의 원소 V_m 는 식 (5)와 같다. Z 는 N 차 정사각행렬이고, m 행 n 열의 원소 Z_{mn} 은 식 (6)과 같다. G 는 $e^{-jk_0 |\vec{r} - \vec{r}'|} / 4\pi |\vec{r} - \vec{r}'|$ 이다. I 는 계수 a_n 으로 이루어진 열벡터이다. 식 (4)~(6)은 EFIE 식 (1)에 관한 행렬식이고, 동일한 과정으로 MFIE 식 (2)에 관한 행렬식을 얻을 수 있으며, EFIE와 MFIE의 결과들의 선형결합으로 CFIE에 관한 행렬식을 얻을 수 있다. MFIE와 CFIE 공식의 조건수(Condition number)가 EFIE에 비하여 낮아 수렴속도가 빠르나, 폐곡면 형태의 산란체에만 적용가능하다^[2].

일단 $I=Z^{-1}V$ 를 구하면, 계수 a_n , 전류 \vec{J} , 산란파를 순차적으로 얻을 수 있다. $I=Z^{-1}V$ 를 계산하는 방법에는 직접법과 반복법이 있다. 직접법의 연산량은 $\mathcal{O}(N^3)$ 로 느리나, Z행렬을 LU 분해하면 다른 \vec{V} 벡터가 주어져도 해를 빠르게 얻을 수 있는 장점이 있다.

반복법에는 CGS(CGGradient Squared), BiCGstab(Bi-CG Stabilized), GMRES(Generalized Minimal Residual) 등이 있다^[17].^[18] 대표적인 CG방법의 수도(pseudo)코드는 <표 1>에 보여진다.

위 과정에서 가장 연산량이 높은 부분은 벡터-행렬 곱셈 $Zd_{(i)}$ 이며, 연산량은 $\mathcal{O}(N^2)$ 이다. 따라서 반복횟수가 l 일 경우에 반복법의 연산량은 $\mathcal{O}(lN^2)$ 이고, 일반적으로 $l < N$ 이기 때문에 직접법보다 수렴속도가 빠르다.

행렬-벡터 곱셈에서 m 번째 원소 $V_m = \sum_{n=1}^N Z_{mn}I_n$ 는 N 개의 기저함수들이 m 번째 테스트함수에 미치는 영향을 의미한다. N 개의 기저함수들과 N 개의 테스트함수들 사이의 상호작용들을 모두 고려하면 매우 많은 연산이 소요되기 때문에, FMM 방법은 이들을 그룹화하여 모든 원소들 사이의 상호작용 대신에 그룹들 사이의 상호작용을 고려한다. 즉, 하나의 메시가 모든 메시에 영향을 주므로, 메시지를 그룹화하여 상호작용 수를 줄일 수 있다. 이때, G 계산 시 소스와 관측점을 분리해야 그룹화가 가능하므로, addition theorem을 사용하여 소스와 관측점을 분리한다^[15].

기저함수 \vec{r} 이 크기가 d 이고 중심이 b 인 그룹 B 에 속하

<표 1> CG 방법의 수도코드

$$\begin{aligned}
 d_{(0)} &= r_{(0)} = V - ZI_{(0)} \\
 (\text{For } i &= 0, 1, 2, \dots) \\
 \alpha_{(i)} &= \frac{r_{(i)}^H r_{(i)}}{d_{(i)}^H Z d_{(i)}} \\
 I_{(i+1)} &= I_{(i)} + \alpha_{(i)} d_{(i)} \\
 r_{(i+1)} &= r_{(i)} - \alpha_{(i)} Z d_{(i)} \\
 \beta_{(i+1)} &= \frac{r_{(i+1)}^H r_{(i+1)}}{r_{(i)}^H r_{(i)}} \\
 d_{(i+1)} &= r_{(i+1)} + \beta_{(i+1)} d_{(i)}
 \end{aligned}$$

고, 테스트함수 \vec{r} 이 동일한 크기의 중심이 a 인 그룹 A 에 속한다고 가정하자. [그림 2]에서 보듯이 기저함수와 테스트함수 사이의 거리 $R = |\vec{r} - \vec{r}'|$ 을 $|\vec{r}_{ra} + \vec{r}_{br}'|$ 로 표시하고, 그룹 간 거리가 $|\vec{r}_{ab}| \gg |\vec{r}_{ra} + \vec{r}_{br}'|$ 인 경우, 식 (6)에서 사용되었던 $G = e^{-jk_0 R} / 4\pi R$ 를 addition theorem을 사용하여 식 (7)과 같이 나타낼 수 있다.

$$G \approx \oint_1 \exp(-jk_0 \hat{k} \cdot \vec{r}_{ra}) \cdot T_L \cdot \exp(-jk_0 \hat{k} \cdot \vec{r}_{br}') dS \quad (7)$$

$$T_L = \frac{k_0}{(4\pi)^2} \sum_{i=0}^L (-j)^{i+1} (2i+1) h_i^{(2)}(k_0 |\vec{r}_{ab}|) P_i(\hat{k} \cdot \hat{r}_{ab}) \quad (8)$$

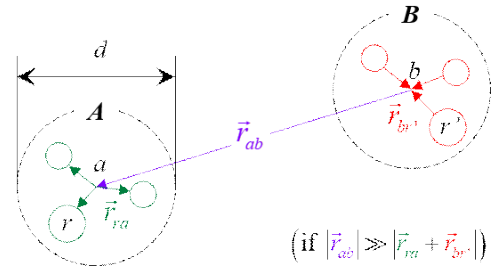
$$L = k_0 |\vec{r}_{ra} + \vec{r}_{br}'| + 6(k_0 |\vec{r}_{ra} + \vec{r}_{br}'|)^{1/3} \quad (9)$$

여기서 $h_i^{(2)}$ 는 i 차 제 2종 구면한켈함수이고, P_i 는 i 차 르장드르함수이고, L 은 무한급수의 절단 차수이며, 식 (9)와 같이 계산된다. 식 (7)의 피적분함수가 의미하는 바는 다음과 같다. 그룹 B 에 속한 기저함수들은 각자의 위치에서 중심 b 로 위상 천이(phase shift)를 통하여 모인다(agggregation 과정). 그리고 이들의 영향이 그룹 간 상호작용을 통하여 그룹 B 에서 그룹 A 로 전달된다(translation 과정). 그리고 그룹 A 에 속한 테스트함수들은 중심 a 에서 각자의 위치로 위상 천이를 통하여 퍼진다(disaggregation 과정). 마지막으로 피적분함수를 단위원 상에서 적분하는데(integration 과정), 적분기호 하한의 1은 단위원을 의미한다.

식 (7)을 식 (6)에 대입하면 아래의 식 (10)을 얻는다.

$$Z_{mn} \approx \oint_1 R \cdot T_L \cdot T dS \approx \sum_{p=1}^{K_a} \sum_{q=1}^{K_b} w_p w_q R \cdot T_L \cdot T \quad (10)$$

$$T = (\vec{T} - \hat{k} \hat{k}) \cdot \iint_S \vec{f}_n \exp(-jk_0 \hat{k} \cdot \vec{r}_{br}') d\vec{r}' \quad (11)$$



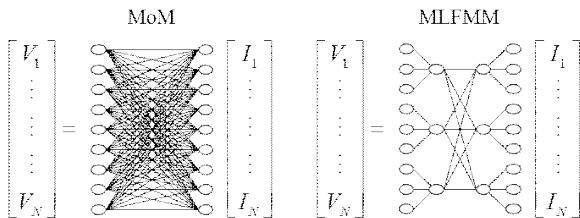
[그림 2] 기저함수들과 테스트함수들의 그룹 간 상호작용

$$R = (\vec{T} - \hat{k}\hat{k}) \cdot \iint_S \vec{f}_m \exp(-jk_0 \hat{k} \cdot \vec{r}_{ra}) d\vec{r} \quad (12)$$

여기서 T , T_L , R 은 각각 radiation, translation, receiving 함수라 부르며, T_L 은 식 (8)과 같다. \vec{T} 는 단위행렬(identity matrix)이다. 식 (10)의 단위원 상의 적분을 수행하기 위하여, 고도 각 θ 와 방위각 ϕ 의 방향으로 각각 K_θ 와 K_ϕ 개수로 차분화한 후, 가중치 w_p 와 w_q 를 사용하여 가우시안 적분을 수행한다. 통상적으로 $K_\theta = L$, $K_\phi = 2L$ 의 샘플링 개수가 사용된다^{[14],[15]}. 그룹의 개수가 $M \ll N$ 일 경우, FMM 방법은 MoM 방법의 연산량 $O(N^2)$ 를 $O(M^2)$ 으로 낮출 수 있으나, 그룹의 개수가 늘어나면, 다시 연산량이 증가한다.

이러한 문제를 해결하기 위하여, MLFMM 방법은 FMM 방법의 그룹화과정을 계층적으로 적용한다. 우선 산란체를 큰 그룹들로 분할하고, 멀리 떨어진 큰 그룹들에 한하여 그룹 간 상호작용을 고려한다. 가까이 위치한 큰 그룹들은 다시 작은 그룹들로 분할하고, 멀리 떨어진 작은 그룹들에 한하여 그룹 간 상호작용을 고려한다. 가까이 위치한 작은 그룹들은 다시 더욱 작은 그룹들로 분할하고, 이 과정을 그룹의 크기가 0.25λ 정도에 이를 때까지 반복한다. 마지막까지 남은 가까이 위치한 그룹들에 속한 기저함수들과 테스트함수들 간의 상호작용은 MoM 방법을 통하여 계산한다.

위와 같은 방식으로 MLFMM 방법은 $O(N \log N)$ 으로 연산량을 낮출 수 있다. 이를 위하여, 기저함수들과 테스트함수들이 어느 하위 그룹에 속하고, 그 그룹들은 또 어느 상위 그룹에 속하는가에 관한 정보가 필요하다. 이는 일반적으로 팔진트리(octree) 형태로 저장되는데, 각 레벨마다 $O(N)$ 의 연산량이 수행되고 레벨의 깊이가 $O(\log N)$ 이므로, 총 연산량 $O(N \log N)$ 을 얻는다. [그림 3]은 MoM 방법과 MLFMM 방법에서 수행되는 행렬-벡터 곱셈의 차이점을 도식화한 것으로, 훨씬 적은 연산량이 사용됨을 확인할 수 있다.



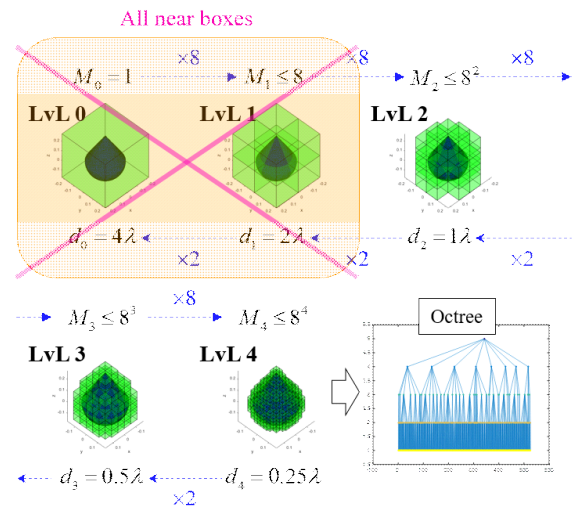
[그림 3] MoM 방법과 MLFMM 방법의 행렬-벡터 곱셈

2-2 Octree

Octree를 만드는 과정은 [그림 4]와 같다. 먼저 산란체를 포함하는 큰 박스를 설정한다(레벨 0). 이 박스를 가로, 세로, 그리고 높이 방향으로 2등분하여 8개의 작은 박스들을 만든다(레벨 1). 산란체를 포함하지 않는 박스는 제외한 후, 남은 박스들을 각각 8등분하여 작은 박스들을 만든다(레벨 2). 이 과정을 박스의 크기가 0.25λ 정도에 이를 때까지 반복하면, [그림 4] 우측 하단의 octree 데이터 구조를 얻는다. 여기서 M_i 는 레벨 i 의 박스 개수이고, d_i 는 레벨 i 의 박스 크기이다. 레벨 0과 레벨 1의 박스들은 모두 근접한 영역에 위치하기 때문에, 사용되지 않는다. 생성된 모든 박스들은 자신의 박스 번호, 박스의 레벨, 박스의 좌표, 부모(parent) 박스의 번호, 자식(child) 박스들의 번호 등의 정보를 포함하는 구조체로 정의될 수 있다. 이후로는 [그림 4]에서 생성된 octree 기반으로 기본 알고리즘을 설명한다.

2-3 Aggregation

일단 octree가 생성되면, 기저함수 레벨에서 [그림 4]의 최하위 박스 레벨인 레벨 4까지 aggregation 과정을 수행한다. 우선 [그림 3]의 행렬 Z 에 곱해지는 벡터 I 를 기저함수에 관한 radiation 함수 식 (11)과 곱한다. 이때 각각의 radiation 함수마다 이후의 적분 과정에서 필요한 레벨 4의 샘플 개수인 K_4 개를 계산해야 하며, 따라서 총 계산량은 $N \times K_4$ 이다.



[그림 4] Octree를 만드는 과정과 생성된 octree

여기서 $K_4 = L_4 \times 2L_4$ 이고, $L_i = k_0 d_i + 6(k_0 d_i)^{1/3}$ 이다. 레벨 4에서 같은 박스에 속한 결과들끼리 서로 합치는(merge) 과정을 통하여, $N \times K_4$ 개의 데이터로부터 $M_4 \times K_4$ 개의 데이터를 얻는다. 여기서 M_4 는 레벨 4의 박스 개수이다.

레벨 3에서 필요한 샘플 개수 K_3 은 K_4 보다 많다. 이를 위하여 위에서 서술한 방식처럼 $N \times K_3$ 개의 데이터를 다시 계산하고, 이로부터 $M_3 \times K_3$ 개의 데이터를 얻을 수 있으나, 다소 비효율적이기 때문에, 일반적으로 보간법이 사용된다. 보간법 방식으로는 크게 전역(global) 보간법과 국소(local) 보간법이 있으며, 전자의 경우에는 FFT를 활용하여 속도향상을 기대할 수 있으나, 이를 위해서는 θ 방향과 ϕ 방향의 샘플들이 균등간격으로 배치되어야 한다^[15]. 후자의 경우에는 르장드르 보간법이 주로 사용된다^[10].

보간법을 사용하여 레벨 4에서 얻은 $M_4 \times K_4$ 개의 데이터로부터 $M_4 \times K_3$ 개의 데이터를 얻는다. 그리고 레벨 3에서 같은 박스에 속한 결과들끼리 서로 합치는 과정을 통하여, $M_4 \times K_3$ 개의 데이터로부터 $M_3 \times K_3$ 개의 데이터를 얻는다. 합치는 과정에서, 레벨 4 박스들의 중심에서 레벨 3 박스들의 중심으로 향하는 위상 천이가 수행된다. 레벨 3에서 레벨 2까지 도달하는 과정은 이와 동일하며, 앞서 언급한 바와 같이 레벨 0과 레벨 1은 고려되지 않는다.

Aggregation 과정에서는 각 레벨마다 $M_i \times K_i$ 개의 데이터가 계산된다. 여기서 $M_i \sim O(N)$ 이고, $K_i = 2L_i^2 \sim (k_0 d_i)^2$ 이며, $d_4 = 0.25\lambda$ 이므로 $K_i \sim O(1)$ 이다. 여기서 d_4 는 최하위 박스 크기이다. 레벨이 증가할수록 M_i 가 늘어나고 K_i 가 감소하여 서로 상쇄되기 때문에, 각 레벨의 연산량은 $O(N)$ 으로 유지되며, 이는 레벨의 깊이 $O(\log N)$ 과 곱하여 aggregation 과정의 총 연산량 $O(N \log N)$ 을 얻는다.

2-4 Translation

Aggregation 과정을 통하여 [그림 3]의 벡터 I 가 곱해진 기저함수들이 octree 모든 레벨의 박스들로 전달되었고, 이제 박스들 사이의 상호작용들을 고려할 차례이다. 이는 최상위 레벨인 레벨 2부터 순차적으로 진행되며, 그 과정은 다음과 같다. 우선 레벨 2 박스들의 radiation 함수와 이로부터 멀리 떨어진 레벨 2 박스들로 향하는 translation 함수를 서로 곱하며, 연산량은 $M_2 \times M_2 \times K_2$ 이다. 레벨 2에서 서로

가까이 위치하여 고려되지 않은 박스들에 한하여, 그들의 자식 박스들만 레벨 3에서 다시 고려한다.

여기서 근거리에 위치한 박스들은 통상적으로 해당 레벨에서 자신의 박스와 바로 인접한 박스들로 이루어진 27개의 박스들로 정의되며, 이외의 박스들은 원거리에 위치한 것으로 간주된다. 따라서 레벨 3에서 translation 과정은 최대한 $M_3 \times (27 \times 8 - 27) \times K_3$ 번 수행되고, $M_3 \times 27 \times K_3$ 번은 레벨 3에서 근거리 상호작용으로 간주되어 그들의 자식 박스들만 레벨 4에서 다시 고려한다. 레벨 4에서도 동일한 과정이 수행되며, 레벨 4에서도 근거리에 위치한 박스들은 그들에게 속한 기저함수들과 테스트함수들 사이의 상호작용을 MoM 방법으로 계산한다.

Translation 과정에서는 최상위 레벨에서 $M_2 \times M_2 \times K_2$ 개의 연산이 수행되고, 나머지 레벨에서는 $M_i \times O(1) \times K_i$ 개의 연산이 수행된다. $M_i \sim O(N)$ 이고, $K_i \sim O(1)$ 이기 때문에, 각 레벨의 연산량은 $O(N)$ 으로 유지되고, 총 연산량 $O(N \log N)$ 을 얻는다.

Translation 과정에 사용되는 식 (8)의 저장을 위한 메모리 절약을 위하여 여러 가지 방법들이 사용된다. 한 가지 방법은 각 레벨 K_i 개의 샘플들을 aggregation 과정과 유사하게 보간법을 통하여 구하는 것이다^[14]. 다른 방법은 식 (8)은 그룹들 사이의 벡터 \vec{r}_{ab} 에만 의존하고, 각 레벨에는 최대 316 가지 경우의 벡터 \vec{r}_{ab} 들이 존재하므로, 메모리 사용량을 알 수 있다^[15]. 또한 식 (8)을 접근적으로 전개한 후 GO 성분인 1차 항만을 고려하는 방법도 있다^[20].

2-5 Disaggregation

Translation 과정을 통하여 [그림 3]의 벡터 I 는 모두 그룹들 사이에서 전달되었고, 다음은 테스트함수들로 전달될 차례이다. 레벨 2에서 translation을 거쳐 전달받은 $M_2 \times K_2$ 개의 데이터들은 receiving 함수 식 (12)와 곱해져야 한다. 이를 위하여 aggregation 과정과 동일한 방식으로, $N \times K_4$ 개의 receiving 함수들을 동일한 상위 그룹에 속한 하위 그룹들끼리 서로 모으고, 그들의 샘플들을 보관하는 과정을 반복하여 $M_2 \times K_2$ 개의 데이터를 만든 후에, translation 결과들과 곱할 수 있다. 이 방식은 비록 그 정확도가 높으나, 최종적으로는 어차피 octree 하위 레벨로 내려와야 하는데, 부가적인

상위 레벨로 올라가는 상황이 발생한다^{[10],[15]}.

Receiving 함수들을 상위 그룹들로 모으고, up-sampling한 후에 translation 결과들과 곱하는 것이 아니라, translation 결과들을 down-sampling하고 하위 그룹들로 분배한 후에 receiving 함수들과 곱하는 방법도 있다^{[13],[14]}. 이 방식은 부가적인 과정이 발생하지 않지만, 그 정확도가 다소 떨어진다. 이들을 수식으로 정리하면 식 (13), (14)와 같다.

$$\sum_{p=1q=1}^{K_p^{i-1} K_q^{i-1}} w_p^{i-1} w_q^{i-1} \{W_{i \rightarrow (i-1)} \cdot R(\hat{k}_{pq}^i)\} \cdot A_L(\hat{k}_{pq}^{i-1}) \quad (13)$$

$$\sum_{p=1q=1}^{K_p^i K_q^i} w_p^i w_q^i R(\hat{k}_{pq}^i) \cdot \tilde{W}_{(i-1) \rightarrow i} \cdot A_L(\hat{k}_{pq}^{i-1}) \quad (14)$$

$$\sum_{p=1q=1}^{K_p^i K_q^i} R(\hat{k}_{pq}^i) \cdot W_{i \rightarrow (i-1)}^T \cdot \{w_p^{i-1} w_q^{i-1} A_L(\hat{k}_{pq}^{i-1})\} \quad (15)$$

여기서 $W_{i \rightarrow (i-1)}$ 는 샘플들을 레벨 i 에서 레벨 $i-1$ 로 보간하는 행렬이고, $\tilde{W}_{(i-1) \rightarrow i}$ 는 레벨 $i-1$ 에서 레벨 i 로 down-sampling하는 행렬이며, $A_L(\hat{k}_{pq}^{i-1}) = T_L(\hat{k}_{pq}^{i-1}) \cdot T(\hat{k}_{pq}^{i-1})$ 이다. 식 (13), (14)에는 receiving 함수들과 곱한 후 단위 원 상에서 수행되는 적분과정까지 포함되어 있다.

식 (13), (14)의 단점들을 보완한 것이 식 (15)이며, antepolation이라 알려져 있다^{[10],[15]}. 여기서 $W_{i \rightarrow (i-1)}^T$ 는 샘플들을 레벨 i 에서 레벨 $i-1$ 로 보간하는 행렬의 전치(transpose) 행렬이다. 이 방법은 보간법과 적분의 순서가 바뀌어 수행되며, 앞서 설명한 두 가지 방법 중, 전자의 방법처럼 그 정확도가 높고, 후자의 방법처럼 부가적인 과정이 발생하지 않는다. Disaggregation 과정은 aggregation 과정과 순서만 다를 뿐 유사한 과정을 거치기 때문에, 동일한 $O(M \log M)$ 의 연산량을 갖는다.

2-6 Integration & Near-Field

Disaggregation 마지막 과정에서 데이터의 샘플 개수에 관한 적분이 수행되며, 이는 식 (10), 그리고 식 (13)~(15)과 같고, 그 연산량은 $O(N)$ 이다. 일반적으로 고도각 θ 방향으로는 L 차 가우시안 구적법이 사용되고, 방위각 ϕ 방향으로는 $(2L+1)$ 차 심슨 구적법이 사용된다^{[10],[15]}.

적분의 결과로 획득한 N 차 열벡터는 기저함수들과 테스트

함수들의 원거리 상호작용만 고려된 것이며, octree 최하위 레벨에서 근거리에 위치한 박스들의 기저함수들과 테스트함수들 사이의 상호작용 결과를 더해 주어야 한다. 이들은 통상적인 MoM 방법 식 (6)으로 계산되며, 그 연산량은 $O(N)$ 이다. 여기서 MoM 방법으로 고려되는 상호작용의 개수는 행렬 Z 의 원소 개수 N^2 에 비하여 훨씬 작기 때문에, 이를 N 차 정사각행렬의 형태로 저장하면, 대부분이 0인 희소행렬이 된다. 따라서 기저함수들과 테스트함수들의 근거리 상호작용은 희소행렬 데이터 구조를 사용하여 저장하는 것이 효과적이며, 여기에는 COO(Coordinate list) 방법 또는 CRS(Compressed Row Storage) 방법 등이 사용된다^[21].

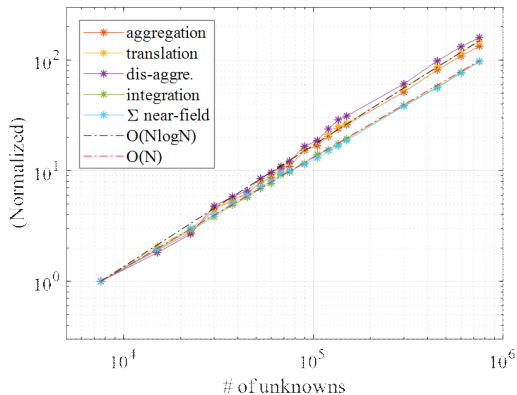
COO 방법은 0이 아닌 원소 값, 열 번호, 그리고 행 번호의 쌍(tuple)을 저장하는 형태로, 0이 아닌 원소의 개수를 Q 라 할 때, $3Q$ 의 메모리를 사용한다. CRS 방법은 0이 아닌 원소의 값, 열 번호, 그리고 행마다 0이 아닌 원소의 개수를 누적시킨 값을 저장하는 형태로, $2Q + (N+1)$ 의 메모리를 사용한다. 여기서 N 은 행 개수이며, $Q \gg N$ 일수록 CRS는 COO보다 많은 메모리를 절약할 수 있다. 그러나 COO에 비하여 부가적인 계산이 수반된다.

Aggregation, translation, disaggregation, integration, 그리고 near-field 상호작용들까지 모두 마치면, 최종적으로 [그림 3]의 N 차 열벡터 V 를 얻는다. 연산량은 각각 $O(M \log M)$, $O(M \log M)$, $O(M \log M)$, $O(N)$, 그리고 $O(N)$ 이므로, 총 연산량은 $O(M \log M)$ 이고, 이는 통상적인 행렬-벡터 곱셈의 연산량 $O(N^2)$ 에 비하여 현저히 적은 값이다. MLFMM 방법의 반복법 내부의 행렬-벡터 곱셈은 이와 같은 과정을 반복하여 수행된다.

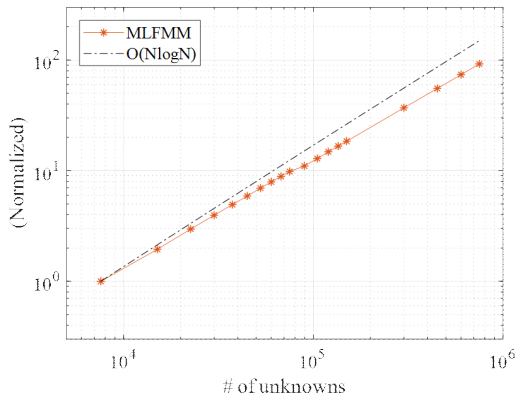
[그림 5]는 지금까지 서술한 MLFMM 방법의 수행결과를 나타낸다. (a)는 aggregation에서 near-field 상호작용에 이르는 각 과정들의 연산시간을 $O(M \log M)$ 과 $O(N)$ 의 그래프와 비교한 결과이다. (b)는 MLFMM 방법의 메모리 사용량을 $O(M \log M)$ 과 비교한 결과이다. 비교를 위하여 시간과 메모리는 정규화되었고, 그림에서 보듯이 결과들은 $O(M \log M)$ 또는 $O(N)$ 의 경향성을 잘 따르고 있다.

III. 결 론

산란해석 수치해석방법에 주로 사용되는 MLFMM 방법을



(a) 연산시간



(b) 메모리 사용량

[그림 5] MLFMM 방법의 수행결과

MoM 방법과 비교하여 살펴보았다. MoM 방법은 고주파 근사법보다 정확한 결과를 제공하지만, 산란체의 전기적 길이가 커지면, 연산량이 $O(N^2)$ 으로 증가하기 때문에, 계산에 어려움이 존재한다. 반면에, MLFMM 방법은 MoM 방법의 반복법에서 수행되는 행렬-벡터 곱셈을 기저함수들과 테스트함수들의 그룹화, aggregation, translation, disaggregation, integration, 그리고 near-field 상호작용의 과정을 통하여 그 연산량을 $O(N \log N)$ 까지 낮출 수 있어, MoM 방법보다 더 큰 산란체의 해석이 가능하다. 현재 MLFMM 방법은 다양한 전처리기법들을 사용하여 반복법의 반복횟수를 낮추고^{[22],[23]}, 병렬처리기법 등을 사용하여 연산시간을 최소화하는 추세로 발전하고 있다^{[24],[25]}.

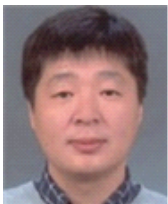
참 고 문 헌

- [1] M. Skolnik, *Introduction to Radar Systems*, 2nd Edition, McGraw-Hill, 1980.
- [2] C. Balanis, *Advanced Engineering Electromagnetics*, 2nd Edition, Wiley, 2012.
- [3] H. Ling, R. Chou, and S. Lee, "Shooting and bouncing rays: Calculating the RCS of an arbitrary shaped cavity", *IEEE Trans. Antennas Propagat.*, vol. 37, no. 2, pp. 194-205, Feb. 1989.
- [4] F. Obelleiro, J. Rodriguez, and R. Burkholder, "An Iterative physical optics approach for analyzing the electromagnetic scattering by large open-ended cavities", *IEEE Trans. Antennas Propagat.*, vol. 43, no. 4, pp. 356-361, Apr. 1995.
- [5] B. Keller, "Geometrical theory of diffraction", *J. Opt. Soc. Amer.*, vol. 52, pp. 116-130, Feb. 1962.
- [6] R. G. Kouyoumjian, P. H. Pathak, "A uniform geometrical theory of diffraction for an edge in a perfectly conducting surface", *IEEE Trans. Antennas Propagat.*, vol. 62, no. 11, pp. 1448-1462, Nov. 1974.
- [7] P. Ufimtsev, *Fundamentals of the Physical Theory of Diffraction*, John Willey & Sons, 2007.
- [8] P. Johansen, "Uniform physical theory of diffraction equivalent edge currents for truncated wedge strips", *IEEE Trans. Antennas Propagat.*, vol. 44, no. 7, pp. 989-995, Jul. 1996.
- [9] R. F. Harrington, *Field Computation by Moment Methods*, IEEE, 1996.
- [10] W. C. Gibson, *The Method of Moments in Electromagnetics*, 2nd Edition, CRC Press, 2014.
- [11] V. Rokhlin, "Rapid solution of integral equations of classical potential theory", *Journal of Computational Physics* vol. 60, no. 2, pp. 187-207, 1985.
- [12] N. Engheta, W. D. Murphy, and V. Rokhlin, "The fast multipole method(FMM) for electromagnetic scattering problems", *IEEE Trans. Antennas Propagat.*, vol. 40, no. 6, pp. 634-641, Jun. 1992.
- [13] O. Ergul, "Fast multipole method for the solution of electromagnetic scattering problems", Master's Thesis, Dept. Elect. and Electron. Eng., Bilkent Univ., Ankara, Turkey, 2003.

- [14] O. Ergul, L. Gurel, *The Multilevel Fast Multipole Algorithm (MLFMA) for Solving Large-scale Computational Electromagnetics Problems*, John Wiley, 2014.
- [15] W. Chew, J. Jin, E. Michielssen, and J. Song, *Fast and Efficient Algorithm in Computational Electromagnetics*, Artech House, 2001.
- [16] M. Rao, D. R. Wilton, and A. W. Glisson, "Electromagnetic scattering by surfaces of arbitrary shape", *IEEE Trans. Antennas Propagat.*, vol. AP-30, no. 3, pp. 409-418, May 1982.
- [17] Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd Edition, SIAM, 2003.
- [18] B. Richard, R. Barrett, W. B. Michael, F. C. Tony, D. James, D. June, D. Jack, E. Victor, P. Roldan, R. Charles, and H. Van der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*, SIAM, 1994.
- [19] J. Shewchuk, "An introduction to the conjugate gradient method without the agonizing pain", Aug. 1994.
- [20] R. J. Burkholder, D. Kwon, "High-frequency asymptotic acceleration of the fast multipole method", *Radio Science*, vol. 31, no. 5, pp. 1199-1206, Oct. 1996.
- [21] W. H. Press, *Numerical Recipes: The Art of Scientific Computing*, 3rd Edition, Cambridge University Press, 2007.
- [22] J. Lee, Z. Zhang, and C. Lu, "Incomplete LU preconditioning for large scale dense complex linear systems from electromagnetic wave scattering problems", *Journal of Computational Physics*, vol. 185, pp. 158-175, 2003.
- [23] J. Lee, Z. Zhang, and C. Lu, "Sparse inverse preconditioning of multilevel fast multipole algorithm for hybrid integral equations in electromagnetics", *IEEE Trans. Antennas Propagat.*, vol. 52, no. 9, pp. 2277-2287, Sep. 2004.
- [24] T. Topa, A. Noga, and A. Karwowski, "Adapting MoM with RWG basis functions to GPU technology using CUDA", *IEEE Antennas and Wireless Propag. Lett.*, vol. 10, pp. 480-483, Sep. 2011.
- [25] J. Guan, S. Yan, and J. Jin, "An OpenMP-CUDA implementation of multilevel fast multipole algorithm for electromagnetic simulation in Multi-GPU computing systems", *IEEE Trans. Antennas Propagat.*, vol. 61, no. 7, pp. 3607-3616, Jul. 2013.

≡ 필자소개 ≡

고 일 석



1992년 2월: 연세대학교 전자공학과 (공학사)
 1994년 2월: 연세대학교 전파공학과 (공학석사)
 1996년 4월: LG전자 연구원
 2004년 4월~2010년 3월: 인하대학교 정보통신
 대학원 교수
 2010년 3월~현재: 인하대학교 전자공학과 교수
 [주 관심분야] Theoretical Electromagnetics, Numerical Analysis, Wave Propagation through Complex Media

이 현 수



2012년 8월: 인하대학교 전자공학과 (공학사)
 2014년 8월: 인하대학교 전자공학과 (공학석사)
 2014년 9월~현재: 인하대학교 전자공학과 박사
 과정
 [주 관심분야] Computational Electromagnetics