

# Çok Seviyeli Hızlı Çokkutup Yönteminin Büyük Elektromanyetik Saçılım Problemlerinin Çözümü için Paralleştirilmesi

Özgür Ergül, Ali Rıza Bozbulut ve Levent Gürel<sup>1</sup>  
Bilkent Üniversitesi  
Elektrik ve Elektronik Mühendisliği Bölümü  
Bilkent, Ankara  
lgurel@ee.bilkent.edu.tr

**Özet:** Üç boyutlu karmaşık geometrilere sahip büyük saçılım problemlerinin hassas doğruluk kriterlerine bağlı hızlı çözümleri için çok seviyeli hızlı çokkutup yöntemi (MLFMM: multi-level fast multipole method) paralelleştirilmiştir. Geliştirilen verimli algoritmaların sayesinde yöntemin yerel ağ sistemiyle birbirlerine bağlanmış işlemciler üzerinde paralel olarak çalışabilmesi sağlanmıştır. Bu sayede bilgisayar olanaklarının izin verdiği sınırlar dahilinde daha önce çözümü mümkün olmayan problemlerin çözümleri hassas doğruluk kriterleriyle gerçekleştirilmiştir. Paralel algoritmanın 6 işlemcili bir grup üzerinde kullanılmasıyla elde edilen 270,000 ve 370,000 bilinmeyenli iletken küre problemlerinin çözümleri örnek olarak gösterilmiştir.

## 1. Giriş

Üç boyutlu karmaşık geometrilerle ilgili saçılım problemlerinin elektromanyetik benzetim ortamında yapılan çözümlerinde, geometrilerin büyüklüğünün artmasıyla birlikte bilinmeyen sayısı da hızlı bir şekilde artmaktadır. Bu nedenle, bilgisayar donanımlarının izin verdiği olanaklar dahilinde çok büyük problemlerin çözümü, hızlı ve verimli algoritmalara rağmen, olanaksız hale gelmektedir. Gerçek hayatta karşılaşılan bu problemlerin çözümü için paralel bilgisayar kümeleri üzerinde çalışabilen algoritmalara ihtiyaç duyulmaktadır. Bu bağlamda, verimli bir algoritma olan çok seviyeli hızlı çokkutup yönteminin [1] paralelleştirilerek çok büyük elektromanyetik problemlerin çözümünde kullanılması gündeme gelmiştir. Bu paralelleştirme işleminin verimli bir şekilde yapılması için işlemcilerle dağıtılan iş yükünün ve işlemciler arasındaki haberleşmelerin dengeli ve planlı bir şekilde yapılması gerekmektedir [2]. Böylece paralel donanımlardan elde edilen faydalar maksimum seviyeye getirilebilir ve çok büyük problemlerin çözümü de öngörüldeği biçimde mümkün olabilir.

## 2. Çok Seviyeli Hızlı Çokkutup Yönteminin Paralleştirilmesi

Saçılım problemlerinin iteratif çözümüne dayalı bir yöntem olan hızlı çokkutup yöntemi (FMM: Fast Multipole Method) [3], matris-vektör çarpımlarının hassas doğruluk kriterleriyle hızlı bir şekilde yapılmasını sağlamaktadır. Bu yöntemde sadece bazı (yakın-alan) etkileşimler momentler metodundaki (MOM: method of moments) gibi doğrudan hesaplanmaktadır. Öte yandan, geri kalan diğer etkileşimlerin hesaplanması doğrudan değildir; temel ve test fonksiyonlarının gruplandırılmasını takip eden üç aşamadan oluşur: Demetleme (aggregation), öteleme (translation) ve dağıtma (disaggregation).

Çok seviyeli hızlı çokkutup yöntemi bu işlemlerin birden çok seviyede yapılması ve grupların gruplandırılması fikrine dayalı olup matris-vektör çarpımlarının çok daha hızlı bir şekilde yapılmasına olanak sağlamaktadır. Bu yöntemle yapılan matris-vektör çarpımları için öngörülen CPU zamanı  $O(N \log N)$ 'dir ve buna ek olarak yakın-alan etkileşimleri dışındaki empedans matrisi elemanlarının doğrudan hesaplanıp bellekte tutulmasına gerek kalmamasından dolayı bellek kullanımı da  $O(N \log N)$  ile sınırlıdır. Böylece, bilgisayar donanımlarının izin verdiği olanaklar dahilinde büyük saçılım problemlerinin çözümü mümkün olmaktadır. Öte yandan, daha büyük problemlerin çözümü için, bu hızlı algoritmaların paralel bilgisayar kümeleri üzerinde çalışabilecek hale getirilmesi ve böylece bu çeşit donanımların sağladığı olanaklardan da faydalanılması gerekmektedir.

Çok seviyeli algoritmada yapılan demetleme ve öteleme işlemlerin paralelleştirilmesi ayrıntılı olarak ele alınmıştır.

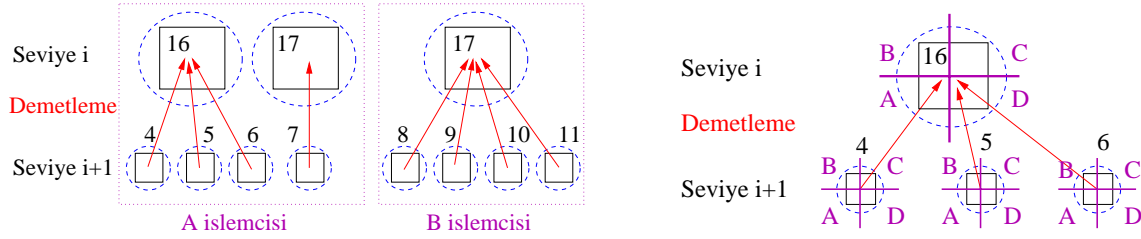
<sup>1</sup> Bu çalışma, Türkiye Bilimler Akademisi (LG/TÜBA-GEBİP/2002-1-12), TÜBİTAK (103E008), ASELSAN ve SSM tarafından desteklenmektedir.

1) *Demetleme işleminin paralelleştirilmesi*: Demetleme işleminde her seviyedeki grupların merkezinde bulunan ışınımların kendilerini içeren ana grupların merkezine gönderilmesi gerekmektedir. Küçük gruplardan büyük gruplara doğru yapılması gereken bu işlem şu denklemle gösterilebilir:

$$\mathbf{F}^{C'_i}(\hat{\mathbf{k}}) = \sum_{C'_{i+1} \in G(C'_i)} \beta_{C'_i C'_{i+1}} \bar{\mathbf{I}}_{i,i+1} \cdot \mathbf{F}^{C'_{i+1}}(\hat{\mathbf{k}}). \quad (1)$$

Bu denklemde  $i$  seviyesinde bulunan  $C'_i$  grubunun ışınımının kendi alt gruplarının ( $G(C'_i)$ ) ışınımlarından nasıl elde edilebileceği gösterilmiştir.

Demetleme işlemi için iki farklı paralelleştirme algoritması öngörülmüştür. Birinci algoritmada her seviyedeki gruplar işlemcilerle yaklaşık olarak eşit bir biçimde dağıtılmıştır. Şekil 1(a)'da algoritmanın genel olarak nasıl çalıştığı şematik olarak gösterilmektedir. Bu örnekte  $i+1$  seviyesinde bulunan 4-7 numaralı gruplar A işlemcisine ve 8-11 numaralı gruplar B işlemcisine verilmiştir. Böylece 4-6 numaralı grupları kapsayan 16 numaralı veli grup (üst grup) da A işlemcisine verilmiş ve bu grupla ilişkili olan demetlemenin tek bir işlemci üzerinde hızlı bir şekilde yapılması sağlanmıştır. Benzer bir durum 17 numaralı grubun B işlemcisine verilmesinde de vardır. Öte yandan, bu grubun altında bulunan 7 numaralı grup A işlemcisindedir. Bu yüzden, 17 numaralı grup için A işlemcisinde de yer açılması gerekmektedir.



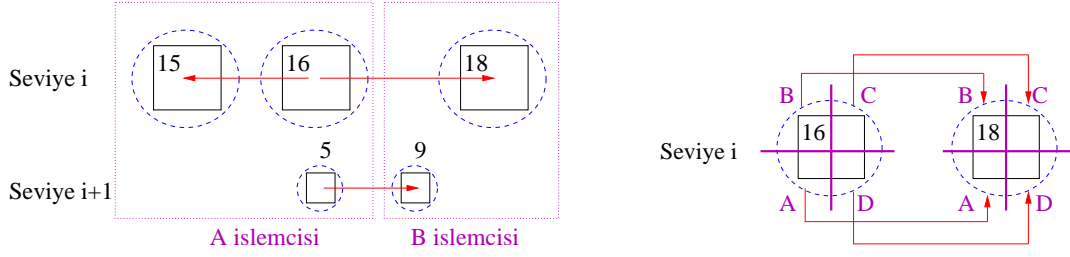
Şekil 1. (a) Birinci demetleme algoritması, (b) ikinci demetleme algoritması.

İkinci demetleme algoritmasında ise her işlemci her grubun belli bir kısmına sahiptir. Şekil 1(b)'de verilen örnekte  $i+1$  seviyesinde 3 grup ve  $i$  seviyesinde bu grupları içeren ana grup gösterilmiştir. Her grubun ışınımını dört işlemci (A-D) paylaşmıştır. Bu durumda herhangi bir demetleme işleminde bu işlemcilerin hepsi de görev almaktadır. İki seviye arasındaki aradeğerlemeler (interpolasyon) düşünüldüğünde, böyle bir demetleme algoritmasında pek çok haberleşmenin gerektiği açıktır. Örneğin, A işlemcisi 16 numaralı grupta kendine ait kısımları doldururken alt gruplardan (4-6) sadece kendi sahip olduğu bölgelere değil, diğer işlemcilerin sahip olduğu bölgelere de kısmen ihtiyaç duymaktadır. Durum böyle olunca, ikinci algoritmanın birinciye göre çok daha verimsiz ve gereksiz olduğu düşünülebilir. Öte yandan bu algoritma, aşağıda açıklanacağı gibi öteleme aşamasında birinci algoritmaya göre daha verimli olmaktadır.

2) *Öteleme (Translation)*: Çok seviyeli hızlı çokkutup yönteminde her seviyede yapılan öteleme işleminde her grubun daha önce belirlenmiş ve kendisine uzak diğer bazı gruplara ışınımlarını göndermesi gerekmektedir. Ötelemenin yapıldığı gruplar birbirlerine uzak olan fakat veli grupları birbirlerine yakın olan (değen) gruplardır ve ötelemenin sonucunda test grubu  $C'_i$ 'nin merkezine gelen dalga

$$\mathbf{F}_T^{C'_i}(\hat{\mathbf{k}}) = \sum_{P(C'_i) \in N(P(C'_i))} \sum_{C'_i \in F(C'_i)} T_L(k, |\mathbf{r}_{cc'}|, \hat{\mathbf{r}}_{cc'} \cdot \hat{\mathbf{k}}) \mathbf{F}^{C'_i}(\hat{\mathbf{k}}) \quad (2)$$

olarak yazılabilir. Öteleme işleminde kullanılacak algoritmalar demetleme algoritmasına doğrudan bağlantılıdır. İlk olarak birinci demetleme algoritmasını ele aldığımızda, Şekil 2(a)'da gösterilen durum ortaya çıkmaktadır. 15 ve 16 numaralı gruplar için gösterildiği gibi, her işlemci öteleme işlemlerinden bazılarını kendi içerisinde gerçekleştirebilir. Öte yandan, 16 ve 18 numaralı gruplar ile alt seviyede 5 ve 9 numaralı gruplar arasındakiler gibi pek çok öteleme işleminde ise işlemciler arasında haberleşme gerekmektedir. Bu haberleşmelerin getirdiği yük özellikle büyük gruplara gidildikçe programın verimliliği açısından oldukça önemli olmaya başlamaktadır.

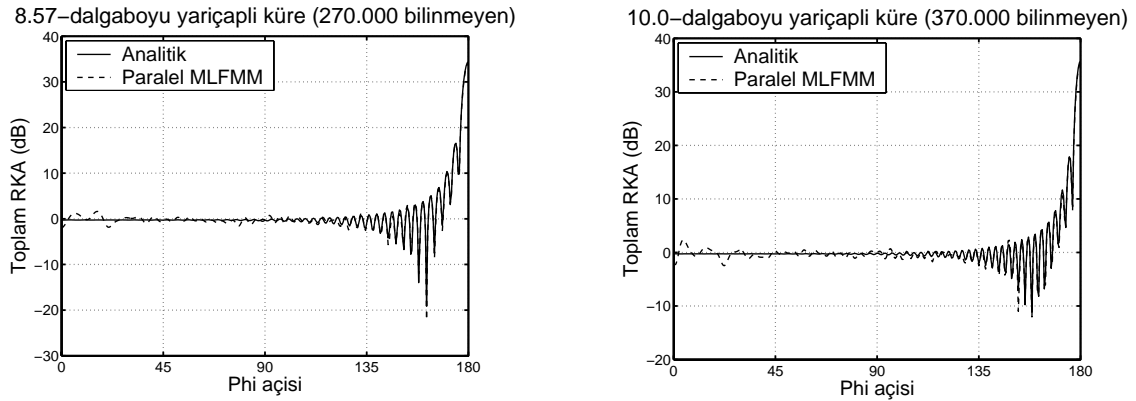


**Şekil 2.** (a) Birinci algoritmadaki öteleme işlemleri, (b) ikinci algoritmadaki öteleme işlemleri.

İkinci demetleme algoritmasında ise durum oldukça farklıdır. Şekil 2(b)'de gösterildiği gibi bu algoritmadaki öteleme işlemleri için herhangi bir haberleşmeye ihtiyaç duyulmamaktadır. Çünkü her işlemci sadece kendi sahip olduğu bölgeleri kullanarak bu işlemi tamamlayabilir. Durum böyle olunca, öteleme işlemlerinde ikinci algoritmanın daha verimli olduğu anlaşılmaktadır. Verimli bir paralelleştirme için her iki algoritmanın da uygun bir biçimde kullanılması düşünülmüştür. Bu amaçla, küçük grupların bulunduğu ve ötelemeden çok demetleme işleminin baskın olduğu alt seviyelerde birinci algoritmanın, büyük grupların bulunduğu ve bu yüzden de ötelemelerin daha kritik olduğu üst seviyelerde ise ikinci algoritmanın kullanılması daha uygun olmaktadır.

### 3. Sonuçlar

İki büyük saçılım problemi için paralel çok seviyeli hızlı çokkutup yöntemiyle elde edilen radar kesit alanı (RCS: radar cross section) sonuçları Şekil 3'te gösterilmiştir. Şekil 3(a)'da 8.57 dalgaboyu ve Şekil 3(b)'de 10 dalgaboyu yarıçapına sahip iletken kürelerin  $x$  yönünden gelen ve elektrik alanı polarizasyonu  $y$  yönünde olan düzlemsel dalgalarla aydınlatılmasından elde edilen radar kesit alanları  $x-y$  düzlemi üzerinde sunulmuştur. Bu problemlerden ilki 270,000, ikincisi ise 370,000 bilinmeyenli olup yerel ağ sistemiyle birbirine bağlanan ve MPI arayüzünün kullanıldığı 6 kişisel bilgisayarın (PC) üzerinde çözülmüştür. Her iki çözüm için de kullanılan bellek miktarı işlemci başına 850 MB'den azdır ve yine her iki çözüm de bir saatin altında tamamlanabilmiştir. Çözümlerin doğruluğunu gösterebilmek için Şekil 3'te hesaplanan değerlerle birlikte analitik değerler de verilmiştir. Her iki çözümde de elde edilen değerlerin gerçek değerlerle tutarlı olduğu gözükmektedir.



**Şekil 3.** (a) 8.57 dalgaboyu, (b) 10 dalgaboyu iletken kürelerin  $x$  yönünden gelen düzlemsel dalgalarla aydınlatılmasıyla  $x-y$  düzleminde elde edilen radar-kesit-alanı değerleri.

### Kaynaklar

- [1] Lu C.-C. ve Chew W.C., "Multilevel fast multipole algorithm for electromagnetic scattering by large complex objects," IEEE Trans. Antennas Propagat., cilt 45, no. 10, s. 1488-1493, Ekim 1997.
- [2] Chew W. C., Jin J.-M., Michielssen E., ve Song J., ed., Fast and Efficient Algorithms in Computational Electromagnetics. New York, IEEE Press, 1998.
- [3] Coifman R., Rokhlin V., ve Wandzura S., "The fast multipole method for the wave equation: a pedestrian prescription," IEEE Ant. Propag. Mag., cilt 35, no. 3, s. 7-12, Haz. 1993.