Yüzey İntegral Denklemleri için Eksik LU Öniyileştiricileri†

Tahir Malas[1] ve Levent Gürel[1, 2]
[1]Elektrik ve Elektronik Mühendisliği Bölümü
[2]Bilişimsel Elektromanyetik Araştırma Merkezi (BiLCEM)
Bilkent Üniversitesi, Ankara, 06800
E-posta: tmalas@ee.bilkent.edu.tr, lgurel@bilkent.edu.tr

Özet: *Saçılım ve ışınım problemlerinde kullanılan, yüksek güvenilirliğe ve düşük hesaplama karmaşıklığına sahip olan çok seviyeli hızlı çokkutup yönteminin etkili olarak kullanılabilmesi, uygun öniyileştiricilerin geliştirilmesine bağlıdır. Bu çalışmada, eksik LU (ELU) öniyileştiricilerinin ardışık bilgisayar ortamlarında etkili çalıştığı, kapalı yüzeye sahip geometrilerin çözüm zamanını iki katına kadar, açık yüzeye sahip geometrilerin çözümünü ise 8 katına kadar hızlandırdığı gösterilmiştir.*

1. Giriş

Karmaşık yapıya sahip üç boyutlu yapıların elektromanyetik saçılım ve ışınım problemlerinin hızlı ve güvenilir çözümlerinde, yapılan hata payını kontrol altında tuttuğundan ve düşük hesaplama karmaşıklığından ötürü çok seviyeli hızlı çokkutup yöntemi (ÇSHÇY) (MLFMA: multilevel fast multipole algorithm) yaygın olarak kullanılmaktadır. Momentler metodu ile ayrıklaştırılan yüzey integrallerinin iteratif çözümünde, ÇSHÇY matris-vektör çarpımını $O(n\log n)$ zamanda yaparak çok büyük problemlerin makul bir zaman içinde çözülmesine imkan sağlamıştır [1]. Ancak, bu yöntemin etkili olabilmesi için iterasyon sayılarının önyileştirmeyle azaltılmasına ihtiyaç bulunmaktadır.

Öniyileştirme için tersi kolaylıkla alınabilen ama aynı zamanda çözülecek olan sistem matrisine mümkün olduğunca yaklaşan bir $\overline{M}$ operatörü seçilerek, $\overline{Z}\cdot a = v$ yerine $\overline{M}^{-1}\cdot\overline{Z}\cdot a = \overline{M}^{-1}\cdot v$ (soldan önyileştirme), veya $(\overline{Z}\cdot\overline{M}^{-1})\cdot(\overline{M}\cdot a) = v$ (sağdan öniyileştirme) çözülebilmektedir. ÇSHÇY, sistem matrisini $\overline{Z} = \overline{Z}^{YA} + \overline{Z}^{UA}$ şeklinde ayırmaktadır ve bu ayrışımdaki uzak alan etkileşimlerini içeren $\overline{Z}^{UA}$ hafızada tutulmamaktadır. Dolayısıyla, öniyileştiriciler seyrek yakın alan matrisi $\overline{Z}^{YA}$ kullanılarak oluşturulmaktadır. Bu bağlamda, yakın alan matrisinin tam çözümünü içeren $(\overline{Z}^{YA})^{-1}\cdot\overline{Z}\cdot a = (\overline{Z}^{YA})^{-1}\cdot v$ öniyileştirmesi, yakın alan matrisinin LU çarpanlarına ayrışımı esnasında çarpanların seyreklik özelliğini kaybetmesinden dolayı makul olmamaktadır. Ancak, LU çarpanlarının dolmasına sebep olan bazı dolumlar (fill-in) ihmal edilebilir ve yakın alan matrisi $\overline{Z}^{near} \approx \overline{L}\cdot\overline{U}$ şeklinde yaklaşık olarak çarpanlarına ayrılabilir. Daha sonra, iterasyonlar sırasında öniyileştirme geri ve ileri çözümlerle sağlanabilmektedir. Bu öniyleştiriciye eksik LU (ELU) (ILU: incomplete LU) ismi verilmiştir [2].

Dolumların nasıl ihmal edildiğine bağlı olarak çeşitli ELU öniyileştiricileri ileri sürülmüştür. Yaygın olarak kullanılan ELU(0), çarpanlarda sıfırdan farklı elemanları asıl matristeki kadar ve aynı yerlerde olacak şekilde tutmaktadır. ELU(0) simetrik kesin artı (symmetric positive definite) matrislerde etkili çalışmakta, kötü koşullu ve tekile yakın matrislerde ise genelde başarısız olmaktadır. Bu tür matrisleri içeren sistemlerin çözümünde ELUT (ILUT) öniyileştiricisi daha güvenilir bulunmuştur [3]. ELUT, eksik LU çarpanlarındaki sıfır olmayan elemanları sayısal değerlerine bakarak atmakta, fakat bununla beraber hafıza gereksinimi de kontrol altında tutabilmektedir.

Bu çalışmada, eksik ELU öniyileştiricileri arasından, ELU(0)'ın, yüzey integral denklemi formülasyonlarından birleşik alan integral denklemi (BAİD) için uygun olduğu gösterilmiştir. ELU(0) sıklıkla kullanılan blok-diyagonal

† Bu çalışma, TÜBİTAK (105E172), Türkiye Bilimler Akademisi (LG/TÜBA-GEBİP/2002-1-12), ASELSAN ve SSM tarafından desteklenmektedir.

öniyileştiricisi ve denektaşı (benchmark) olarak kullanılan yakın alan matrisinin tam çözümü ile kıyaslanmıştır. Sonuç olarak, özellikle gerçek problemlerde, ELU(0) blok-diyagonale nazaran çok daha iyi çalışmıştır. Üstelik, ELU(0)'ın iterasyon sayıları, yakın alan matrisinin tam çözümüne çok yakın çıkmıştır.

Yüksek derecede belgisiz (indefinite) ve tekile yakın matrisler üreten elektrik alan integral denklemi (EAİD) içinse, ILU(0)'dan daha güvenilir olan ELUT (ILUT) öniyileştiricisi kullanılmıştır. Bu çalışmada, ELU(0)'la aynı derecede hafıza kullanarak, ELUT'nin pek çok EAİD problemlerinde çalıştığı gösterilmiştir. Bununla birlikte, bazı problemlerde ELUT ile yakınsamanın elde edilemediği, hatta ilginç bir şekilde çözümün zorlaştığı görülmüştür. ELU öniyileştiricilerinin karşılaştığı en büyük sorun, çarpanlarına ayırma işlemi esnasında, küçük eksenel elemanlardan (pivots) kaynaklanan sıfırdan farklı matris elemanlarının kontrolsüzce büyümesi ve kararsız LU çarpanları oluşturmasıdır. Bu durumun meydana gelip gelmediğini anlamak için [3]'te tavsiye edildiği gibi $\left\|(\bar{\mathbf{L}} \cdot \bar{\mathbf{U}})^{-1}\right\|_{\infty}$ için bir üst sınır olan *condest* değerlerine bakılmıştır. *Condest* değeri şu formülle hesaplanmaktadır:

$$\left\|(\bar{\mathbf{L}} \cdot \bar{\mathbf{U}})^{-1} \cdot \mathbf{e}\right\|_{\infty}, \qquad \mathbf{e} = [1,1,\ldots 1]^{T}. \tag{1}$$

Bu şekilde iterasyonlar başlamadan önce, elde edilen ELU öniyileştiricisinin etkili olup olamayacağı anlaşılabilmektedir. Genelde, *condest* değeri 10^{6}'dan büyükse, iteratif çözümde yakınsama elde edilememektedir.

Kararsız ayrışımlar için bir dizi önlemler ileri sürülmüştür [3]. EAİD matris denklemlerinin çözümünde, ucuz fakat etkili bir yöntem olan kolon eksenel döndürülmesi (column pivoting) denenmiş, bunun kararsızlık problemini giderdiği saptanmıştır. Elde edilen öniyileştiriciye ELUTP (ILUTP) ismi verilmektedir.

2. Sayısal Sonuçlar

ELU öniyileştiricilerinin etkinliğini sınayabilmek amacı ile kapalı geometrilerin BAİD ile, açık yüzeye sahip geometrilerinse EAİD ile elde edilmiş sistemleri üzerinde sayısal deneyler yapılmıştır. ELUT'de büyüklüğü 10^{-6}'dan küçük olan elemanlar ihmal edilmiş ve bir matris satırındaki izin verilen en fazla sıfırdan farklı eleman sayısı yakın alan matrisinin ortalamasına eşit tutularak ELU(0)'la yakın miktarda bellek gereksinimi sağlanmıştır.

Kıyaslama amacı ile ELU(0) ve ELUT'ye ek olarak şu öniyileştiriciler gerçekleştirilmiştir:

- **LU:** Yakın alan matrisinin tam olarak çarpanlarına ayırımı ile elde edilmiştir. İterasyon zamanını düşürme bakımından yakın alan matrisi ile elde edilebilecek en kuvvetli öniyileştiricidir. Ancak, hem bellek kullanımı, hem de çözüm zamanı açısından çok pahalı olduğu için pratik kullanımı yoktur. Sınanan matrislerin yakın alan matrisine ne kadar iyi bir yaklaşım sunduğunu anlamak amacı ile denektaşı (benchmark) olarak kullanılmıştır.
- **Blok-diyagonal (BD):** ÇSHÇY yönteminin ağaç yapısı kullanılarak elde edilmiştir. En son seviyedeki kümelerin öz etkileşimlerini ihtiva etmektedir. Yakın alan matrisinin küçük blok-diyagonallerine denk geldiği için tersi kolaylıkla alınmaktadır.
- **Diyagonal (D):** EAİD matrisi çözümlerinde ilginç bir biçimde blok-diyagonal öniyileştiricisi, öniyileştirici kullanmamaya nazaran daha kötü sonuç vermektedir. Bu yüzden ELUT ile karşılaştırabilmek amacıyla sadece yakın alan matrisinin diyagonali öniyileştirici olarak kullanılmıştır.
- **Seyrek yaklaşık ters (SYT):** Daha ziyade paralel gerçekleştirimlerde kullanılan bir öniyileştiricidir. Yakın alan matrisinin tersine doğrudan yaklaşmakta olup, $\left\|\bar{\mathbf{I}} - \bar{\mathbf{Z}}^{YA} \cdot \bar{\mathbf{S}}\right\|_{F}$'nin enküçültmesi ile hesaplanmıştır.

İteratif çözücü olarak GMRES [2] kullanılmıştır. Özellikle EAİD matrislerinin çözümünde GMRES'in diğer Krylov altuzayı çözücülerine kıyasla daha başarılı olduğu gözlemlenmiştir. İterasyonlara ilk tahmin olarak sıfır vektörüyle başlanmış, $\|\mathbf{r}_0\| / \|\mathbf{r}_k\| < 10^{-6}$ olduğunda iterasyon durdurulmuştur. Tüm öniyileştiriciler sağdan uygulanmıştır.

Tablo 1'de kapalı geometrilerin BAİD'le elde edilmiş sistemlerinin anılan öniyileştiricilerle iterasyon sayıları (iter), ELU(0) ve SYT için kurulum zamanları ve çözüm zamanı (kurulum + iterasyon zamanı) saniye cinsinden verilmiştir.

Problem	Bilinmeyen Sayısı	LU	Öİ'siz		BD		ELU(0)			SYT		
		İter	İter	Zaman	İter	Zaman	İter	Yapım	Zaman	İter	Yapım	Zaman
Küre	132,003	29	49	1,103	32	684	29	23	665	29	23,102	23,102
İnce Kutu	147,180	37	158	1,965	106	1,290	45	271	1,025	64	298,479	298,479
Kanat	117,945	31	86	1,110	52	779	32	46	542	37	73,100	73,100
Flamme	78,030	63	229	2,138	115	1,096	66	43	694	76	96,369	96,369
Helikopter	183,546	42	253	7,338	106	3,081	44	145	1,739	61	234,614	234,614

Tablo 1. BAID matrisleri için öniyileştiricilerin karşılaştırılması.

ELU(0) küre geometrisinde iterasyon sayılarını azaltsa da, bu problemin çözümü kolay olduğu için kurulum zamanı çok az olan BD ile çözüm daha kısa sürede elde edilmiştir. Küreye kıyasla daha karmaşık olan diğer geometrilerde ise iterasyon sayısı ve çözüm zamanını önemli miktarda düşürmüştür. Örneğin, karmaşık gerçek yaşam geometrileri olan Flamme [4] ve helikopter geometrilerinde iterasyon sayısı ve çözüm zamanı BD'nin yaklaşık yarısına düşürülmüştür. Ayrıca, elde edilen iterasyon sayıları LU'ya çok yakın çıkmıştır. Öte yandan, SYT karmaşık geometriler için iterasyon sayısını azaltsa da yüksek kurulum maliyetinden dolayı toplam çözüm zamanını düşürememiştir.

EAİD problemlerini içeren çözümler ise Tablo 2'de sunulmuştur. Kapalı yüzeyler BAİD ile çok daha kolay ve hızlı çözülebildiği için EAİD ile açık yüzeye sahip geometrilerin çözümleri karşılaştırılmıştır.

Problem	Bilinmeyen Sayısı	LU	Diyagonal		ELUTP			SYT		
		İter	İter	Zaman	İter	Yapım	Zaman	İter	Yapım	Zaman
Levha	137,792	53	833	16,209	81	661	2,167	92	19,955	21,384
Açık Küp	171,655	332	-	-	376	2,243	9,833	354	207,436	207,436
Açık Prizma	163,871	195	-	-	253	996	6,883	396	57,606	57,606
Yarım Küre	116,596	93	1,052	25,947	110	1,353	3,579	156	22,079	22,079

Tablo 2. EAİD matrisleri için öniyileştiricilerin karşılaştırılması.

EAİD için öniyileştirmenin önemi, Tablo 2'den açıkça anlaşılmaktadır. Açık küp ve açık prizma geometrilerinde kuvvetli bir öniyileştirici olmadan yakınsama gözlenememiş, levha ve yarım küre problemlerinde ise ELUTP çözümü 10 katına kadar hızlandırmıştır ve iterasyon sayıları LU'ya oldukça yakın çıkmıştır. Öte yandan, SYT'nin yüksek kurulum maliyeti ardışık bilgisayarlarda yine ciddi bir dezavantaj oluşturmaktadır.

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] Saad Y., *Iterative Methods for Sparse Linear Systems*. SIAM, Philadelphia, ABD, 2003.

[3] Chow E. ve Saad Y., "Experimental study of ILU preconditioners for indefinite matrices," *J. Comput. Appl. Math.*, cilt 86, no. 2, s. 387-414, 1997.

[4] Gürel L., Bağcı H., Castelli J.-C., Cheraly A. ve Tardivel F. "Validation through comparison: Measurement and calculation of the bistatic radar cross section of a stealth target," *Radio Science*, cilt 38, no. 3, s. 1046-1058, 2003.