II. ULUSAL HAVACILIK VE UZAY KONFERANSI **UHUK-2008-1131**

15-17 Ekim 2008, İTÜ, İstanbul

YAPISAL OLMAYAN VE YARIM BAĞLI HEXAHEDRAL SAYISAL AĞLARIN YAY METODU İLE ENİYİLENMESİ

Özgür Uğraş BARAN*
Havelsan, Ankara

ÖZET

Hexahedral ağlar, sayısal hesaplama yöntemlerinde, çözüm hassasiyeti açısından en etkin ağlar olarak kabul edilmektedirler. Blok-yapısal ağlar, bu ağlar arasında en yaygın kullanılanı olmakla birlikte, karmaşık geometrilerde uygulanmaları son derece zordur. Yapısal olmayan hexahedral ağ üretimi ise genellikle düşük kaliteli karışık hücreli ağlar veya tam-bağlı olmayan ağlarla sonuçlanır. Bu çalışma kapsamında, tam hexahedral, tam bağlı olmayan ağların yay metodu ile eniyilenmesine yönelik bir yöntem geliştirilmiş, üretilen ağlar akışkanlar mekaniği tabanlı uygulamalarda kullanılarak sınanmıştır.

GİRİŞ

Geçen onyıllar içerisinde bilgisayar sistemlerinin katlanarak hızlanması çok büyük ölçekli fiziksel simülasyonların ve kişisel bilgisayarlarda dahi karmaşık modellemelerin yapılabilmesine imkan vermiştir. Fiziksel problemlerin sayısal yöntemlerle çözümleri genellikle sayısal bir ağ üzerinde yapılır. Bu sayısal ağlar, çözüm yapılacak kontrol hacminin, daha küçük ve şekilleri önceden tanımlı alt-hacimlere, yani hücrelere bölünmesiyle meydana getirilir. Ardından, fiziksel fenomeni tanımlayan denklem setlerinin çözümüne yönelik sayısal yöntemlerın, ağ boyunca varolan hücrelerin her biri için ve hücreler arası komşuluk ilişkiler gözönüne alınarak uygulanmasıyla fiziksel problem çözülür.

Akışkanlar mekaniği, yapısal mekanik ve benzeri çok sayıda mühendislik disiplinini ilgilendiren problemlerde, sayısız sayısal çözüm yöntemleri bulmak mümkündür. Yapılan akademik çalışmalar ve onyıllardır süregelen endüstriyel deneyim göstermiştir ki sayısal ağların kalitesi ve karakteristikleri en az sayısal çözüm yönteminin

*Dr., Simülasyon ve eğitim sistemleri- Havelsan A.Ş., E-posta: ubaran@havelsan.com.tr

kendisi kadar, pek çok zaman ise ondan bile daha önemlidir. Literatürde çeşitli ağ türleri ve sınıflandırmaları bulmak mümkündür. Pek çok akademik ya da ticari çözücü sadece belirli türde ağlar üzerinde çalışma yapabilmektedir. Sayısal ağların sınıflandırılmasını kabaca aşağıdaki şekilde yapmak mümkündür.

- Yapısal ağlar, hücre bağlantılarının belirli ve sabit bir düzen içerisinde belirlenmesi ile oluşturulur. Yapısal olmayan ağlarda ise hücre komşulukları ve bağlantıları düzensizdir ve hücreler arasındaki bağlantı bilgileri ayrıca sağlanmak zorundadır.
- Hücre şekli itibarı ile, en yaygın kullanımda olan ağlar, düzgün dörtyüzlü hücrelerden ibaret (tetrahedral) ağlardır. Diğer bir yaygın şekil ise düzgün altıyüzlü (hexahedral) ağlardır. Tetrahedral, hexahedral ve piramit şekilli hücrelerin birleşiminden oluşan ağlar da seyrek olarak kullanılmaktadır.
- Kontrol hacmi yüzey, kenar ve köşeleri dışında tüm hücre yüzeylerinin ve kenarlarının tamamen eş diğer bir veya birkaç hücre yüzü, kenarı ve köşesi ile komşuluğunun tanımlandığı ağlar tam bağlı (conformal) ağlar olarak tanımlanır. Eğer bazı hücre yüz, kenar veya köşeleri bağlantısız ise bu elemanlar asılı yüz/ kenar /köşe, bu tür ağlar ise yarım bağlı (non-conformal) ağlar olarak tanımlanır.

Hexahedral ağlar, tetrahedral ağlara kıyasla aynı hücre sayısı için çok daha hassas ve üç ila on kat daha az hücre ile aynı hassasiyette çözüm verebildikleri için daha verimli kabul edilmektedir [5]. Endüstride en sık kullanılan sayısal hexahedral ağ cinsi blok-yapısal ağlardır. Ancak, bilgisayarların hızlanması ile çok karmaşık geometrilerin modellenmesi gündeme gelince, bu tür ağların önemli bir handikap olduğu ortaya cıkmıştır. Blok-yapısal ağlar, ağın tamamlanması için bir veya birkaç mühendisin uzun ve titiz çalışmasına ihtiyaç duymaktadır. Gelinen noktada, ağ üretimi işlemi sayısal çözümlemeden daha uzun zaman almaya başlamıştır. Örneğin belirtilen yöntemle, cok sayıda soğutma kanalına sahip bir türbin kanatçığı için gerekli ağ oluşturma süreci haftalar, kompresör kademeleri ile birlikte tam bir motorun sayısal ağ ile bölünmesi aylar alabilmektedir.

Hexahedral hücrelerin sağladığı çözüm kalitesinden taviz verilmeden ağ üretim kademesini hızlandırmak için otomatik hexahedral ağ üretimi kavramı ortaya atılmıştır. Ancak, hexahedron, tetrahedron'a göre çok daha karmaşık bağlanabilirlik sınırlandırmalarına ve tam bağlı hücre bölünmesinde aşırı derecede kıstlılığa sahiptir. Bu yüzden tetrahedral ağlar için uygulanan ağ üretim yöntemleri, hexahedral ağlar için uygulanamazlar. Yaklaşık yirmi yıldır yapılan çalışmaların sonunda bulunan yöntemler, ya genel kullanım için çok yetersiz, ya sınırlı bir geometri grubuna uygulanabilir ya da içlerinde büyük oranda diğer tür hücreleri barındıran ağlar üretmektedirler [10, 9].

Bu güne kadar ticari olarak kullanılabilecek genellikteki tek otomatik hexahedral ağ üretme yöntemi, oct-tree yöntemidir. Bu yöntem yukarıda örneklenen geometrileri saatler içerisinde tam hexahedral ve yuzeye tam uyumlu ağ ile kaplayabilmektedir. Ancak, yöntem bunu başarırken, hexahedral ağın tam bağlı olmasını aynı anda sağlayamamaktadır. Yöntemi kullanırken bir potansiyel problem ise geometri sınırına yakın hücrelerin kalitesinin ilk oluşturmada daha düşük olmasıdır.

Bu çalışmada, tam bağlı olmayan, geometri yüzeyine tam uyumlu hexahedral ağlar için ağ kalitesini arttırmaya yönelik bir eniyileme yöntemi geliştirilmiştir. Bu yöntem, ağ kenarlarının fiziksel yaylar ile değiştirilip oluşturulan fiziksel sistemin denge durumuna dek hareket ettirilmesinden ibarettir. Yöntem bahsedilen türdeki ağlar ile sınırlı olmayıp her türlü ağ için kullanılabilir. Yöntem asılı düğümler ve yüzler ile başarı ile çalışabilmekte, çok kademeli ağların kademe ilişkilerine uygun eniyileme yapabilmektedir. Yöntem karmaşık geometrilerin ağlarının eniyilenmesinde kullanılmış, oluşturulan ağların kalite metrikleri sunulmuş ve akışkanlar mekaniği tabanlı uygulamalar ile çözüm kalitesi sağlaması yapılmıştır.

YÖNTEM

Oct-tree yöntemi, bilinen en verimli ve başarılı hexahedral ağ oluşturma yöntemidir. Yöntem, aşağıdaki aşamalardan oluşmaktadır [10, 9];

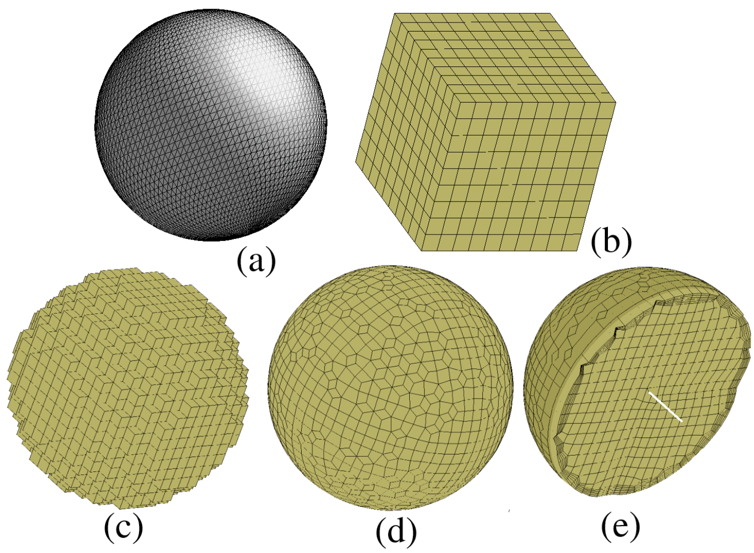
İlk Ağ: Yönteme başlarken, geometri kendisini içine alacak bir ilk ağ ile kaplanır. Bu ilk ağ genellikle kaba, kartezyen bir ağdan ibarettir.

Uyumlandırma: İlk ağın hücreleri belirlenen kriterlere ve geometri özelliklerine göre tam otomatik olarak bölünerek inceltilir.

Tıraşlama: Geometrinin içerisindeki hücreler korunup geometriyi kesen ve dışında kalan hücreler belirlenip atılır.

Yapıştırma: Bu aşamada tıraşlanan ağın en dış yüzeyinde bulunan ağ yüzeyiyle geometri yüzeyi arasına bir ağ hücresi tabakası yerleştirilerek ağın geometriye uyumlu hale gelmesi sağlanır.

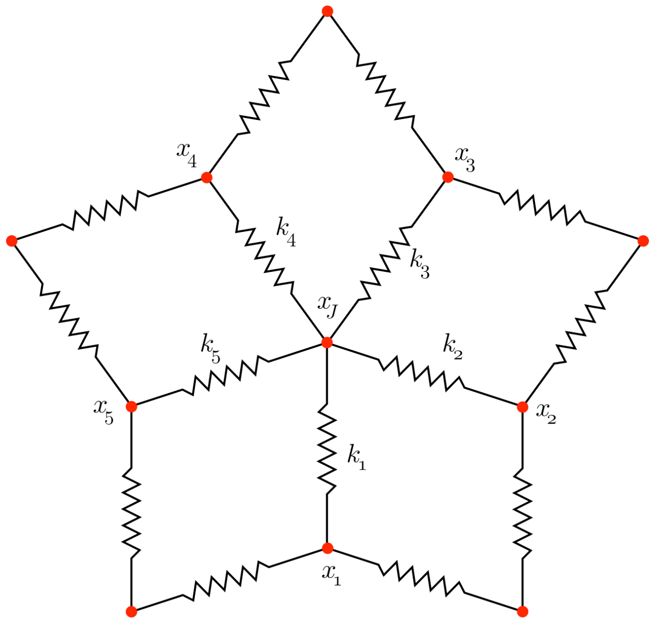
Eniyileme: Yeni oluştulan ağın kalitesinin arttırılması için eniyileme yapılır.



Şekil 1. Oct-tree yöntemi ile eniyileme

Yaylar Yöntemi İle Eniyileme

Yaylar yöntemi ile eniyileme, ağ kenarlarının birbirlerine ağ düğümlerinden bağlı hayali doğrusal yaylar ile değiştirilmesinden oluşur. Bu fikir ilk olarak ortaya Batına tarafında 2 boyutlu hareketli ağları çözümü için ortaya konulmuştur [2, 3]. Ayrıntılı matematiksel analizleri Blom [4]tarafından sunulmuştur. Bu yaylar doğrusal oldukları için Hooke kanununa göre hareket ederler. Dolayısı ile, her yayın kendisine ait bir sertliği $k_{I,J}$ vardır. I ve J ağ düğümlerinin indeksini belirtmektedir. Sonuçta bir ağ düğümü etrafında yer aldığı varsayılan yay sistemi Şekil 2'deki gibidir.



Şekil 2. Bir düğüm etrafında oluşturulan yaylar

Herhangi bir ağ düğümünün ideal pozisyonu, bu nokta etrafındaki yayların enerji minimizasyonu ile çözülür. Buna göre:

$$\min_{x_J \in \mathcal{R}^d} E(x_J) = \min_{x_J \in \mathcal{R}^d} \sum_{I=1}^{n} (x_I - x_J)^2 k_I(x_J) \tag{1}$$

Bu noktanın dengeye geldiği pozisyonda x_J düğümü üzerindeki enerji minimize olacaktır. Bu noktayı bulmak için denklem 1 türevi alınır:

$$\sum_{I=1}^{n} \overrightarrow{F}_I(x_J) = \sum_{I=1}^{n} (x_I - x_J) k_I(x_J) = 0 \tag{2}$$

Hooke kanununa göre, $\overline{x_I x_J}$ kenarı üzerindeki yay kuvveti $F_I(x_J)$, $(x_I - x_J)k_I(x_J)$ olarak verilir, buna göre denklem 2 yeniden yazılırsa,

$$x_J = \frac{\sum_{I=1}^{n} x_I k_I(x_J)}{\sum_{I=1}^{n} k_I(x_J)} \tag{3}$$

Bu denklem sadece tek bir düğüm üzerinde verilmiştir. Bir ağın tüm noktaları üzerinde bu denklem yazılırsa:

$$x_J^{m+1} = x_J^m + \omega \frac{\sum_{I=1}^{n} (x_I^m - x_J^m) k_I(x_J^m)}{\sum_{I=1}^{n} k_I(x_J^m)} \tag{4}$$

burada ω, Gauss-Seidel tipi yaklaşımlı yöntemler için hızlandırma faktörüdür ve $0 < \omega < 2$ aralığında değişmektedir. Bu çalışma için $\omega = 1.2$ değerinin en iyi hızlandırmayı sağladığı görülmüştür. Ancak bu değer yay kuvvetinin başka yaklaşımlarla hesaplandığı, ya da başka tür sayısal ağlar kullanılan çalışmalarda, kısaca problemin çeşidine göre yeniden eniyilenmelidir. Dahası, belirtilen ω aralığı da her türlü problemin çözümü için geçerli olmayabilir.

Burada gösterilen sistemdeki yayların daima gerilme kuvveti altında olduğu kabul edilmektedir. Eğer tüm yay sistemi kendi halinde hareket etmeye bırakılırsa, sistem tüm yay uzunlukları sıfır olacak şekilde bir noktaya doğru çekilecektir. Genişleyen yay formulasyonu ise sistem sonsuza kadar büyüme eğiliminde olacağı için kullanılabilir değildir. Dolayısı ile sınır düğümleri sabitlenmiş, ya da hareketi sınırlandırılmış gergin yaylar sistemi en verimli sistemdir. Daha genelleştirilmiş bir formulasyonda ise, yayların bir baz uzunluğu olduğu kabul edilir ve yayların anlık uzunluklarının bu baz uzunluktan uzun veya kısa olmalarına göre gerilme veya sıkıştırma altında oldukları anlaşılır. Denklem (3), belirli bir yay baz uzunluğuna l_{IJ} göre yazılır ve enerji problemi buna göre çözülürse, aşağıdaki genel formül elde edilir:

$$x_J^{m+1} = x_J^m + \omega \frac{\sum_{I=1}^{n} (x_I^m - x_J^m - l_{IJ}) k_I(x_J^m)}{\sum_{I=1}^{n} k_I(x_J^m)} \tag{5}$$

Çoğu optimizasyon temeli uygulama için baz uzunluğunu sıfır almak yeterlidir. Ancak hareketli sınır problemleri için bu uzunluğun ağ kenarlarının ilk uzunluğu olarak seçilmesi faydalıdır.

Yay yöntemini diğer yöntemlere göre çok daha esnek kılan şey, yay sabitini seçmekteki serbestidir. Ana denklemde yapılacak sayısal stabiliteyi sağlayacak birkaç küçük değişiklikle kullanıcı istediği yöntemle yay sertliğini ayarlayabilir [1, 6, 12]. Örneğin, hareketli ağ ya da hareketli sınır problemlerinde, l_{IJ} ağ kenarının orijinal uzunluğu olarak alınabilir. Böylece ağın ilk eniyilemesi hareket sonunda da korunmuş olur. Düzenli (konformal) ağlarda eniyileme çalışmalarında ise tüm ağ kenarları için 1 ya da daha hızlı ve iyi sonuç vermesi için yay uzunluğunun tersi alınır:

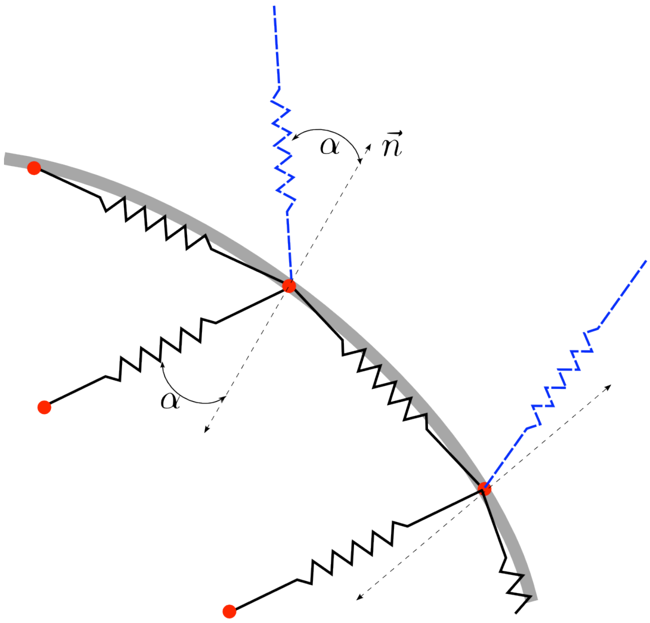
$$k_I(x_J) = \frac{1}{\|x_I - x_J\|} \tag{6}$$

Blum'a göre, bu formülle oluşturulan yay yöntemi gerçekte Laplace yöntemi ile eniyilemeye eşleniktir. Yaratılan sistem çok hızlı ve basit olmasına rağmen oldukça sorunsuz olarak çalışır.

Sınır Şartları

Herhangi bir fiziksel sistemin çözümünde sınır şartlarının doğru belirlenmesi en önemli adımdır. Her ne kadar yay sistemi yapay bir sistem olsa da, geometriye bağlılığı ihlal etmeyen ve istenilen ağ kenar uzunluğu ve kalitesine sahip sayısal ağın elde edilmesi için sınır şartlarının dikkatlice belirlenmesi gerekir. Sınırda yer alan düğümlerin sayısal hacmin yüzeyinde kalmaları sağlanmalıdır, aksi takdirde tüm yay sistemi bir noktaya dek küçülecektir. Ancak, sınır üzerinde bulunan düğümlerin konumunu sabitlemek de çok katı bir sistem yaratıp bu düğümlerin en iyi pozisyonlarına ulaşmalarını engelleyip sınır civarında yer alan ağ hücrelerinin kalitesinin düşmesine yol açar. Oysa, ağ üzerinde yapılacak sayısal çözümün hassasiyeti için en kaliteli hücrelerin sınıra en yakın olanlar olması gerekmektedir.

Bu problemin çözülebilmesi için sadece bir ucu sınır üzerinde yer alan her yay için, yönü bu yayın sınıra bağlandığı noktadaki geometrik yüzeyin normaline göre ayna görüntüsünde olan yeni birer izafi yay eklenir. Sonra bu yeni yüzey yayları sistemi, yukarıda anlatıldığı gibi çözülür. Bu yeni sistemdeki sınır yayları, ilk bulundukları sınır noktasına teğet düzlem üzerinde hareket eder. Sistem çözüldikten sonra yay uçlarının geldiği yer, geometrik yüzeyin çoğunlukla çok yakınında, ama her zaman tam üzerinde değildir. Yeni bulunan sınır düğümlerinin tekrar geometrinin sınırında yer alan en yakın noktaya taşınması gerekmektedir.



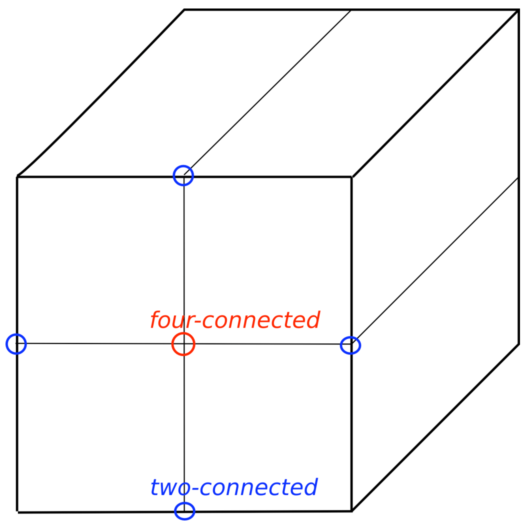
Şekil 3. Yay sistemi için sınır düzeltmesi

Bu işlemi yaparken dikkat edilmesi gereken en önemli nokta, geometrinin topolojisinin tamamen korunmasıdır. Geometri, yüzeyler, yüzeyleri ayıran kenarlar ve kenarları birleştiren köşelerden oluşan bir yapı olarak ele alınır ve bu topolojik yapının eniyileme işlemi sırasında korunmasına azami şekilde önem gösterilmelidir. Bunun için yukarıda anlatılan yeniden yapıştırma işlemi sırasında düğüm noktalarının ait oldukları topolojik yüzey/ kenar/ köşe'yi değiştirmemesi de sağlanmıştır.

Yöntemin Yarım Bağlı Ağlara Uyarlanması

Bu çalışma, ticari bir hexahedral ağ üreteci olan *HEXPRESS* üzerinde gerçekleştirilmiştir. Çok az sayıda bulunan ticari güvenilirliğe sahip bu ağ üreteci, yarım bağlı (non-conformal) hücre bölünmesi ile adaptasyonu yöntemine bağlıdır. Bu işlemin sonunda, asılı düğümler denilen bir ağ yüzeyine veya kenarına bağlanmış sistematik düğümler dizisi ortaya çıkar. Bu tip düğümlerin varlığı yay analojisi ile optimizasyon yöntemi için iki problem ortaya koyar.

Asılı düğümler tam bağlı olanların aksine dengeleyici bir karşı-yay'a sahip olmadıkları için içe doğru çekilip hücrenin yapısını bozmak eğilimindedirler. Bunu çözmek için sınır şartlarında yapıldığı gibi izafi yaylar kullanılabilir. Ancak bu yöntem probleme aşırı bir karmaşıklık getirir. Onun yerine bu çalışmada basit ama etkili geometrik bir yöntem seçilmiştir. Buna göre asılı düğümler, düğümün ebeveyn hücrenin yüzünde veya kenarında olmasına göre iki-bağlı ve dört-bağlı olarak sınıflandırılır. Bu düğümlerin konumları, yay sisteminin çözümü esnasında dondurulur. Ardından, bağlı olma sınıflandırılmasına göre yeni hücre yüzü veya kenarının orta noktasına gelecek şekilde yeniden konumlandırılır.



Şekil 4. İki ve dört-bağlı asılı düğümler

İnceltme operasyonlarının varlığı çözülmesi gereken ikinci problemi ortaya koyar. Bu inceltme operasyonlar, değişik ebadta hücreler yaratır. Normal bir yay eniyilemesi tüm bu hücre kenarlarını birbirine yakın uzunluklara getirmeye çalışır. Bu ise bölünme ile oluşturulmuş hücre büyüklüklerinin bozulması anlamına gelir ve çok düşük kalitede ağların ortaya çıkmasına sebep olur. Bunun için iki yöntem sözkonusudur. Birincisi, her ağ bölünme seviyesinin kendi arasında yay yöntemi ile çözülmesidir. Ancak bu yöntemin bu çalışmayı takip eden çalışmalarda fazladan bir bağlayıcı unsur olduğu ve sistemin esnekliğini gereksiz yere azalttığı görülmüştür. Ayrıca anizotropik bölünmelerde fazladan karmaşıklık ve hatta uygulama kısıtlılığı getirmektedir. Bu yüzden bu çalışmada yay sistemi tüm hacim-içi ağ kenarlarının tek bir sistemde oluşturulmasıyla çözülmüştür.

Kenar boyutunun korunması için daha fiziksel bir yöntem izlenmiştir. Bu ikinci yöntemde, yay sertliği ebeveyn hücre kenarının uzunluğunun bir üst inceltme seviyesindeki kenar uzunluğundan iki kat fazla olacak şekilde ayalanmıştır.

Buna göre, yay yönteminde kullanılan Gauss-Seidel algoritması aşağıdaki gibi kullanılmıştır

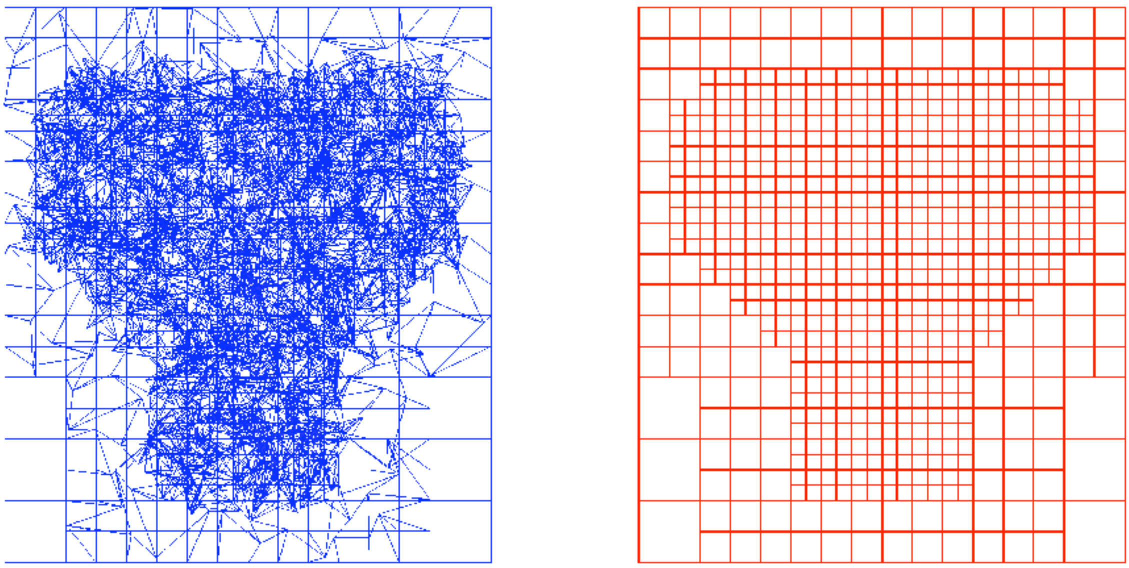
$$x_J^{m+1} = x_J^m + \omega \frac{\sum_{I=1}^{n} W_I (x_I^m - x_J^m) k_I(x_J^m)}{\sum_{I=1}^{n} W_I k_I(x_J^m)} \tag{7}$$

Burada, W_I kenar içi yay sertliği ağırlık faktörüdür ve şu formülle belirlenir:

$$W_I = 2^{2l_I} \tag{8}$$

l_I, kenar inceltme derecesidir. Burada belirtilmesi gereken nokta şudur: Bu formül sadece her kenarın en çok ikiye bölündüğü tam bağlı olmayan çok seviyeli ağlar için geçerlidir ve diğer tür ağlar için ağırlık faktörü güncellenmelidir.

Açıklanan yöntemin geçerliliği öncelikle basit bir ağ üzerinde sınanmıştır. Bunun için üç boyutlu bir geometrinin etrafında üç inceltme operasyonu ile bir ağ oluşturulmuştur. Bu ağ'da 114876 hücre bulunmaktadır. Ağ iç düğümleri gelişigüzel olarak kendi uzunluklarının iki katı mesafeye kadar yerdeğiştirilmiş ve yaylar yöntemi ile $\omega = 1.5$ ve hata toleransı kenar uzunluğunun 10^{-5}'i seçilerek eniyilenmiştir. Tüm eniyileme süreci, 2Ghz pentium sınıfı bilgisayarda 16 sn içinde bitmiştir. Sonuç ise şekilde görüldüğü gibi mükemmele yakındır.

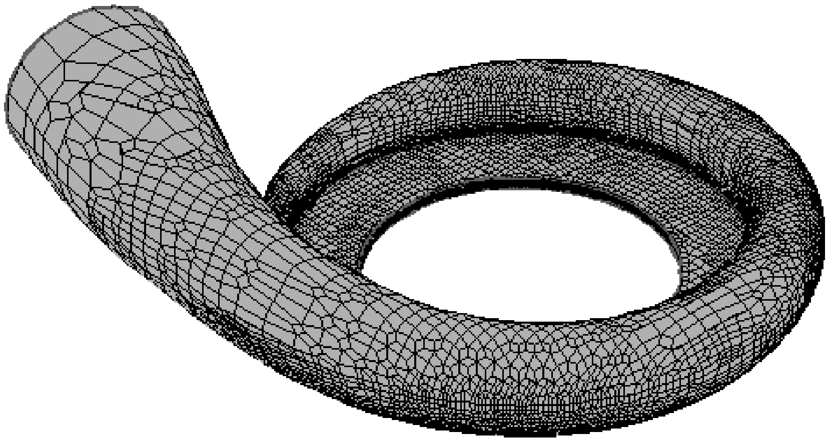


(a) Gelişigüzel hareket ettirimiş ağ mesh (b) Eniyilenmiş ağ

Şekil 5. Oct-tree bir ağın yay sistemi ile eniyilenmesi

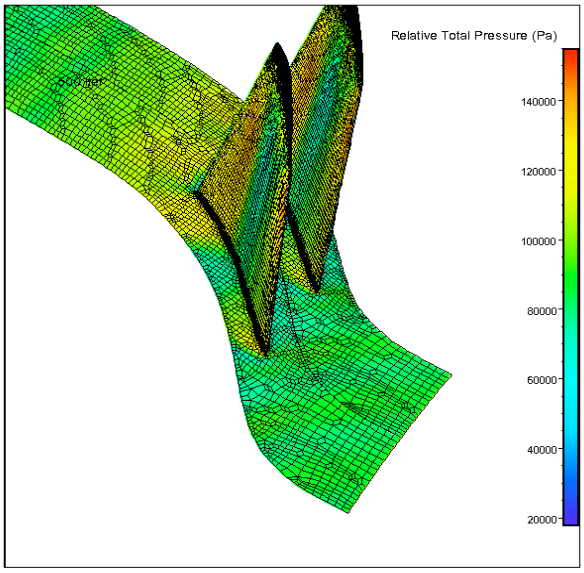
Yöntemin Sınanması

Yukarıda anlatılan yöntem, bir dizi iç ve dış akım kontrol hacimleri için test edilmiştir. Bu amaçla kullanılan ilk kontrol hacmi bir türbin salyangozudur. Bu problemde 79637 ağ hücresi dört inceltme operasyonu sonucunda elde edilmiştir. Yöntemin uygulanmasından sonra çıkış bölgesinde dahi hiç içedönük hücre kalmamış ve ağ kalitesi yükselmiştir. Elde edilen ağ Şekil 6.'da görülebilir.



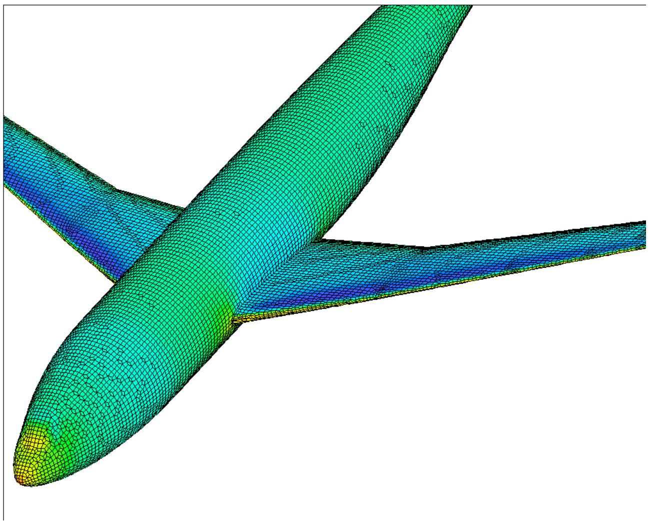
Şekil 6. Salyangoz geometrisi

Diğer bir test de NASA tarafından deneyleri yapılan Rotor37 testi sayısal olarak çözülmüştür. Bu test için yukarıda anlatılan yöntemle eniyilenmiş ve 1.2M hücreden müteşekkil ağ üzerinde 17188.7 RPM hızla dönen ve ilgili raporda [7] verilen sınır şartları kullanılmıştır. Stall durumu için kütle akımı deney sonucunda $m = 0.925 * m_{choke} = 19.36\,kg/s$ olarak verilmiştir. Sayısal çözüm sonucu bulunan değer ise $18.79kg/s$'dir. Kullanılan ağ ve çözüm, Şekil 7'de verilmiştir. Bu çözüm son derece başarılı olarak kabul edilmektedir.

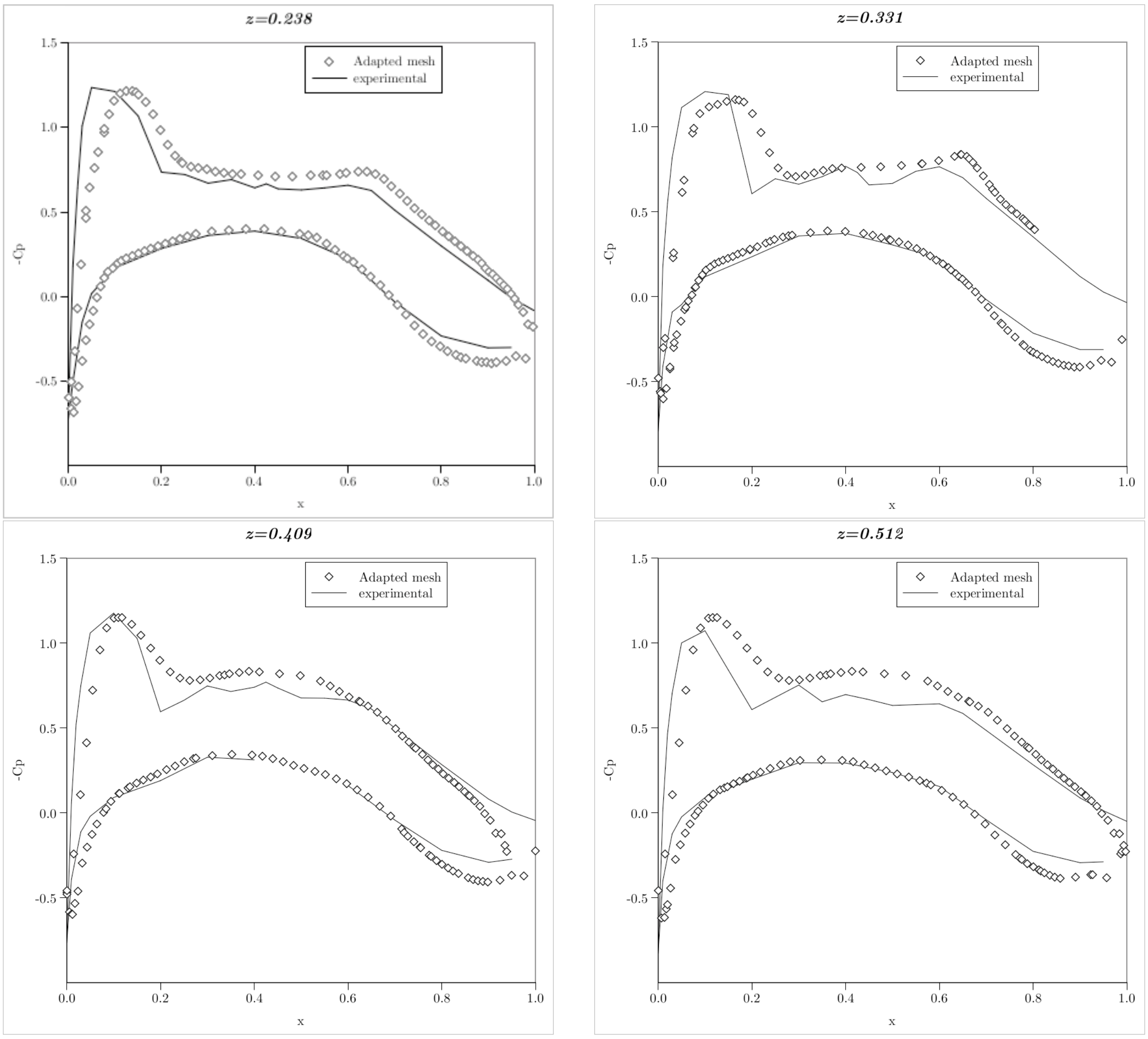


Şekil 7. Rotor37 için göreceli toplam basınç.

İkinci test, standard bir dış akım örnek uygulaması olan DLR-F4 geometrisidir [11]. Bu geometri üzerinde eniyileme yapılmada önce %8 oranında negatif hücre bulunmakta idi. Yaylar yöntemi ile eniyileme sonucunda, 80-90 derece ortogonalite oranına hücrelerin %92'si ulaşmışve hiç negatif hücre kalmamıştır. Bu geometri etrafında 0.92 Mach sayısı için Euler problemi çözümü sağlanmıştır. 102445 adet hücresi bulunan bu ağın eniyilemesi 2 dakikadan az sürmüştür. Euler probleminin sonuçları sağlanan deneysel veri ile uyumludur. Bu testin sonuçları Şekil 8'de görülebilir. Kanat boyunca alınan kesitlerdeki çözümler Şekil 9'da gösterilmiştir ve deneysel sonuçlarla karşılaştırılmıştır [8].



Şekil 7. DLR-F4 ağ ve çözüm



Şekil 8. DLR-F4 çeşitli kesitlerde çözüm

SONUÇ

Bu çalışmada yarım bağlı hexahedral ağlarda yay yöntemi ile eniyileme algoritması sunulmuştur. Bu algoritma, ağ kenarlarının, ağ düğümlerinde birbirine bağlı doğrusal yaylarla değiştirilmesinden ibarettir. Standard algoritmanın, yarım bağlı ağlara uyarlanması sağlanmış ve formülasyon verilmiştir.

Elde edilen algoritma, ticari bir hexahedral ağ üreteci olan HEXPRESS üxerinde uygulanmış ve elde edilen ağlar, sözü edilen türdeki ağları kullanabilen bir çözücüde test edilmiştir. Bu testler sonucunda, yay algoritmasını son derece hızlı çözüm verdiği ve dahası üretilen ağlarda yüksek kalitede hesaplamalı akışkanlar mekaniği sonuçları alındığı görülmüştür.

Kaynaklar

[1] Djaffar Ait-Ali-Yahia, Guido Baruzzi, Wagdi G. Habashi, Michel Fortin, Julien Dompierre, and Marie-Gabrielle Vallet. Anisotropic mesh adaptation: towards user-independent, mesh-independent and solver-independent CFD, Part II: Structured grids. *International Journal For Numerical Methods in Fluids*, 39:657–673, 2002.

[2] J. Batina. Unsteady Euler spring method for three-dimensional unstructured dynamic meshes. *AIAA Paper*, 89-0150, 1989.

[3] J. Batina. Unsteady Euler airfoil solutions using unstructured dynamic meshes. *AIAA Journal*, 28(8):1381–1388, 1990.

[4] Frederic J. Blom. Considerations on the spring analogy. *International Journal For Numerical Methods in Fluids*, 32:647–668, 2000.

[5] A.O Cifuentes and A. Kalbag. A performance study of tetrahedral and hexahedral elements in 3-d finite element structural analysis. *Finite Elements in Analysis and Design*, 12:313–318, 1992.

[6] Julien Dompierre, Marie-Gabrielle Vallet, Yves Bourgault, Michel Fortin, and Wagdi G. Habash. Anisotropic mesh adaptation: towards user-independent, mesh-independent and solver-independent CFD. Part III. Unstructured meshes. *International Journal for Numerical Methods in Fluids*, 39, Issue 8:675 – 702, 2002.

[7] J Dunham. CFD validation for propulsion system components. Technical report, AGARD-AR-355, 1995.

[8] D. W. Levy, T. Zickuhr, J. Vassberg, S. Agraval, R. A. Wahls, J. Pirzadeh, and M. J. Hemsh. Summary of data from the first aiaa cfd drag prediction workshop. Technical report, AIAA Paper 2002-0841, 2002.

[9] Steven Owen. A survey of unstructured mesh generation technology. In *Proceedings of 7th International Meshing Roundtable*, 1998.

[10] Steven J. Owen. *Non-Simplical Unstructured Mesh Generation*. PhD thesis, The Department of Civil and Environmental Engineering Carnegie Mellon University, 1999.

[11] G. Redeker. DLR-F4 wing body configuration. Technical report, AGARD Advisory report No 303 A Selection of Experimental test cases for the validation of CFD Codes, 1994.

[12] A. Tam, Djaffar Ait-Ali-Yahia, M. P. Robichaud, M. Moore, V. Kozel, and Wagdi G. Habashi. Anisotropic mesh adaptation for 3d flows on structured and unstructured grids. *Comput. Methods. Appl. Mech Engrg*, 189:1205–1230, 2000.