

**KİŞİSEL BİLGİSAYARDA  
YAYILMIŞ BELLEK KULLANILARAK  
BÜYÜK BOYUTLU MATRİS İŞLEMLERİNİN GERÇEKLEŞTİRİLMESİ**

**D.Uğur ŞANLI**

**ÖZET**

Bu çalışmada, kişisel bilgisayarlarda standart belleğin yetmediği durumlarda yayılmış bellek (extended memory) kullanılarak büyük boyutlu matris işlemlerinin gerçekleştirilmesi konusu ele alınmıştır. Örnek olarak, ölçü sayısının fazla olduğu ve ölçüler arasındaki korelasyonun göz önünde bulundurulduğu bir dengeleme işleminde, matrislerin boyutunun büyümesi sonucunda standart belleğin aşılması sorunu incelenmiştir.

**ABSTRACT**

This paper discusses how the operations with large dimensional matrices are performed in a PC by using extended memory in case conventional memory is not sufficient. As an example, the situation of exceeding the limits of conventional memory in an adjustment problem where the number of observations are large and correlation between them is considered has been examined.

**1. GİRİŞ**

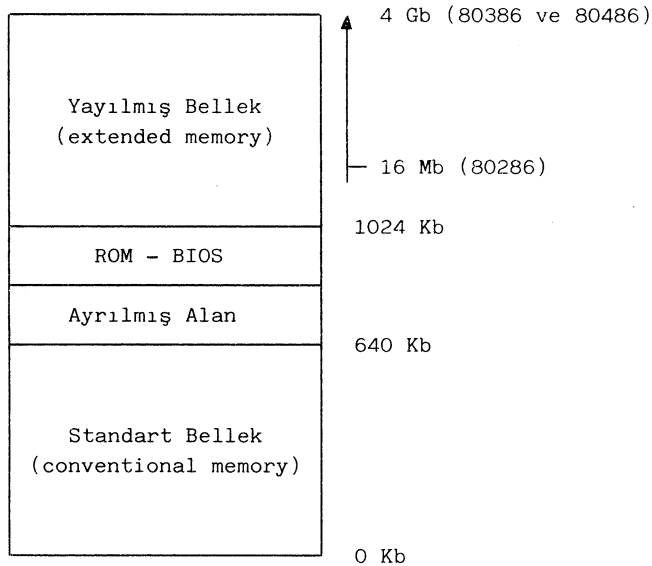
Günümüzde, Jeodezi'deki problemlerin sayısal çözümü için çoğunlukla kişisel bilgisayarlardan yararlanılmaktadır. Kişisel bilgisayarların işletilmesi için yaygın olarak kullanılan DOS işletim sistemi, tüm belleğin (rastgele erişimli bellek/RAM) yalnızca 640 Kb'lık bir bölümünü (conventional memory/standart bellek) kullanabilmekte, bu yüzden veri ve işlem hacmi büyük olan bir çok uygulama gerçekleştirilememektedir. Örneğin, dengeleme hesabı yapılırken, bilinmeyen sayısı arttığında veya gözlem sayısı fazla olup ölçüler arasındaki korelasyonun göz önünde bulundurulması gerektiğinde, hesapta kullanılacak matrislerin boyutu büyümekte ve 640 Kb sınırı aşıldığından bellek yetmezliği sorunu ortaya çıkmaktadır. 500x500 boyutlu bir matrisi ele aldığımızda, eğer tekil duyarlık (single precision) kullanılıyor ise ve bu durumda her bir verinin bellekte 4 byte yer kapladığı göz önünde bulundurulursa, bu matrisin standart bellekte işgal ettiği alan  $500 \times 500 \times 4 = 1$  Mb olacak ve 640 Kb sınırı aşılabacaktır. Görüldüğü gibi sadece bir matris ile kullanılabilir bellek alanı aşılmaktadır, oysa dengeleme hesabı yapılırken bir dizi matris işlemi sonucunda çözüme ulaşılmaktadır.

Bellek yetmezliđi sorunu, UNIX, OS/2 gibi daha az kullanım alanına sahip iřletim sistemleri yardımıyla ya da bilgisayar programlarının, WINDOWS altında alıřabilen üst düzey programlama dillerinde (C, Pascal vb.) hazırlanması ile özöllebilmektedir. Bunun yanında DOS 5.0 ve daha sonraki sürümlerinin, RAM'in 1 Mb dıřında kalan bölümünü (yayılmıř bellek/ek bellek) sabit disk gibi (RAM drive/RAM disk) kullandırma özelliđinden yararlanılarak, bellek yetmezliđi sorununa pratik bir özüm getirilebilir. Hazırlanan programların alıřması sırasında büyük boyutlu matrislere ait veriler, eriřimi sabit diske (hard disk) göre birkaç kat daha hızlı olan bu elektronik bellekte geici olarak depolanabilir ve standart bellek ise yalnız iřlemlerin yapılması amacıyla kullanılır.

Bu alıřmada, ilk önce RAM disk yaratma konusunda kısa bilgiler verilmiř ve yayılmıř bellek kullanımı göz önünde bulundurularak matris iřlemlerini gerekleřtiren algoritmaların nasıl hazırlanacağı açıklanmıřtır. Ayrıca, dengeleme hesabı yapılırken büyük boyutlu matrislerle alıřmanın dođuracağı bellek yetmezliđi sorununun özölmesi konusu incelenmiřtir.

## 2. RAM DİSKİN OLUŐTURULMASI

IBM veya IBM uyumlu kiřisel bilgisayarlarda RAM'in ancak 1 Mb'lık bir bölümü adreslenebilmektedir (Jamsa, 1993). Kiřisel bilgisayarlarda yaygın olarak kullanılan iřletim sistemi DOS, bu 1 Mb'lık alanın standart bellek adı verilen 640 Kb'lık bölümünü kullanabilmektedir (Őekil-1). Bilgisayar aıldıđında DOS eđer CONFIG.SYS dosyasında yüksek bellek veya üst bellek



Őekil-1: IBM bilgisayarda bellek kullanımı (Jamsa, 1993)

alanına yerleşmesi istenmez ise kendisini standart belleğin en alt kısmına yükler. Kullanılacak olan uygulama programı, DOS tarafından geriye kalan bellek alanına yerleştirilir. Eğer programın bellek gereksinimi 640 Kb sınırını aşıyorsa standart belleğe sığmayacaktır. Dolayısıyla, program tarafından üretilen ara değerler geçici olarak yayılmış belleğe aktarılabilir. Yayılmış belleğe ya özel bir yazılım ile ya da RAM disk oluşturularak erişilebilir. RAM disk, sabit disk gibi kullanılabilen elektronik bir bellektir ve aynı zamanda hızı sabit diske göre birkaç kat daha fazladır. Ancak, elektronik bir bellek olduğundan bilgisayar kapatıldığında RAM diskteki veriler kaybolur. Bu yüzden ara değerler arada bir sabit diske aktarılmalıdır. Çalışmanın bundan sonraki bölümünde RAM disk ve yayılmış bellek kavramları eşdeğer anlamda kullanılacaklardır.

RAM diski oluşturmak için CONFIG.SYS dosyasına

```
device = [sürücü: ] / [yol ] /ramdrive.sys [ disk boyutu (Kb) ] /e
```

komut satırı eklenmelidir. Ancak, bundan önce, DOS'un yayılmış belleği kullanabilmesi için yine CONFIG.SYS dosyasında

```
device = c:/dos/himem.sys
```

deyiminin olması gerektiği unutulmamalıdır. Örneğin, 1 Mb'lık bir RAM disk yaratmak istiyorsak komut satırı;

```
device = c:/dos/ramdrive.sys 1024/e
```

olacaktır. Bu konuda daha ayrıntılı bilgi, DOS işletim sistemi kitaplarında bulunabilir (Jamsa, 1993, Korkmaz, 1993).

Bilgisayar açıldığında sabit disk sürücüsü hangi harf ile gösteriliyorsa (çoğu bilgisayarda C:), RAM disk sürücüsü bir sonraki harf ile adlandırılır (D:).

### **3. MATRİS İŞLEMLERİNİ GERÇEKLEŞTİREN ALGORİTMALARIN YENİDEN DÜZENLENMESİ**

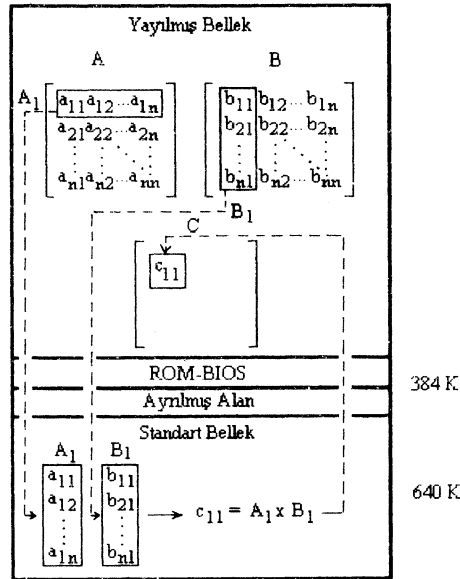
Matris işlemlerini içeren bir bilgisayar programı hazırlanırken, öncelikle kullanılacak matrislerin boyutları tanımlanır. Eğer bir dizi matris işlemi için gerekli olan tüm boyut 640 Kb'ı aşmıyorsa çözüm standart bellek alanı içerisinde gerçekleştirilir. Boyut büyük ise RAM diskten yararlanılacaksa, genelde kullanılan algoritmaların yayılmış bellek kullanımı göz önünde bulundurularak yeniden düzenlenmesi gerekir.

Matrisin elemanları RAM diskte rastgele erişimli (random access) veya ikili (binary) dosyalarda depolanmalıdır. Herhangi bir matris işleminin (çarpma, toplama, çıkarma, evrik ve invers alma) gerçekleştirilebilmesi için veriler bu dosyalardan satır - satır veya satır - sütun düzeninde standart

belleğe alınmalıdır. İşlem gerçekleştirildikten sonra elde edilen ara sonuçlar da doğrudan RAM diske yazdırılmalıdır. Böylece tüm matris yerine, matrisin işleme uğrayacak satır veya sütunları vektörler halinde standart belleğe alınır. Şekil-2'de matris işlemlerinin yayılmış bellek kullanılarak gerçekleştirilmesine örnek olarak, A ve B matrislerinin çarpılması sonucunda elde edilen C matrisinin ilk elemanının bulunması gösterilmektedir. Burada  $A_1$ , A matrisinin birinci satır vektörü,  $B_1$  ise B matrisinin birinci sütun vektörüdür. C matrisinin ilk elemanı  $c_{11}$ 'in hesaplanabilmesi için  $A_1$  ve  $B_1$  standart belleğe yüklenerek

$$A_1 \times B_1 = \sum_{j=1}^n a_{ij} \cdot b_{ji}, \quad i = 1$$

işlemi yapılır. Hesaplanan değer yayılmış bellekte C sonuç matrisi için açılan dosyaya yazdırılır. C matrisinin diğer elemanları da benzer şekilde satır-sütun çarpımları yapılarak bulunur.



Şekil-2: Yayılmış bellek kullanılarak matris çarpımının gerçekleştirilmesi

Toplama işleminde A ve B matrislerinin her ikisinin de birer satırı standart belleğe yüklenir. Satır elemanlarının toplamları burada gerçekleştirilerek, sonuç satır vektörü yayılmış bellekte C matrisi için açılan dosyaya yazdırılır. Bir matrisin evriğinin alınmasında ise yayılmış bellekten standart belleğe aktarılan satır vektörler burada sütun vektöre dönüştürülerek tekrar yayılmış bellekte evrik matris için açılan dosyaya yazdırılırlar.

Matrisin inversinin alınmasında Ayres (1974) ve Mikhail ve Ackermann (1976)'da anlatılan yöntem uygulanmıştır. Bu yöntemle göre büyük matris dört ayrı parçaya ayrılmakta ve

$$AA^{-1} = AB = I$$

yani

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}$$

uyarınca

$$A_{11}B_{11} + A_{12}B_{21} = I$$

$$A_{11}B_{12} + A_{12}B_{22} = 0$$

$$A_{21}B_{11} + A_{22}B_{21} = 0$$

$$A_{21}B_{12} + A_{22}B_{22} = I$$

denklem takımı oluşmaktadır (I = birim matris). Bu denklem takımı çözümlenerek B matrisinin alt matrisleri

$$B_{11} = (A_{11} - A_{12}A_{22}^{-1}A_{21})^{-1}$$

$$B_{12} = -B_{11}A_{12}A_{22}^{-1}$$

$$B_{21} = -A_{22}^{-1}A_{21}B_{11}$$

$$B_{22} = A_{22}^{-1} - A_{22}^{-1}A_{21}B_{12}$$

belirlenir. Görüldüğü gibi burada yalnızca  $A_{22}$  matrisinin inversi standart bellekte alınmakta, diğer işlemler ise yayılmış bellek kullanımı göz önünde bulundurularak düzenlenen algoritmalar ile gerçekleştirilmektedir (Mikhail ve Ackerman, 1976).

#### 4. ZAMAN DİZİLERİNİN ANALİZİ İLE İLGİLİ BİR UYGULAMA

459 ölçü, 23 bilinmeyen oluşun bir zaman dizisi, ölçüler arasındaki korelasyon göz önünde bulundurularak dengelenmiştir. Kullanılan kişisel bilgisayar 80486 DX2-66 model, 4 Mb RAM kapasitelidir. Ölçüler arasındaki korelasyon göz önünde bulundurulduğu için ölçülerin ağırlık matrisi P dolu matris olarak işleme girmekte ve bu matrisin standart bellekte kapladığı alan 640 Kb sınırını aşmaktadır. Dolayısıyla yayılmış bellek kullanımına ihtiyaç duyulmuş ve çözüm için RAM disk'ten yararlanılmıştır.

459 ölçü, 23 bilinmeyen için gerçekleştirilen dengeleme hesabında matris gösterimiyle işlem adımları ve matrislerin boyutları şu şekilde olur:

$$\begin{aligned} \ell + u &= \hat{A}X & \ell &\rightarrow 459 \times 1 \\ \hat{X} &= (A^T P A)^{-1} A^T P \ell & u &\rightarrow 459 \times 1 \\ u &= \hat{A}X - \ell & A &\rightarrow 459 \times 23 \\ & & \hat{X} &\rightarrow 23 \times 1 \\ \hat{\sigma}_0^2 &= u^T P u / n - u & P &\rightarrow 459 \times 459 \end{aligned}$$

Burada  $\ell$  ölçüler,  $u$  ise düzeltmeler vektörü,  $A$  katsayılar matrisi,  $P$  ölçülerin dolu ağırlık matrisi,  $\hat{X}$  bilinmeyenler vektörü,  $\hat{\sigma}_0^2$  ise birim ağırlıklı ölçünün varyansıdır. Görüldüğü gibi  $459 \times 459$  boyutlu  $P$  matrisi, eğer tekil duyarlık kullanılıyorsa bellekte yaklaşık 840 Kb yer kaplamakta, bu da standart bellek alanı sınırını aşmaktadır. Dolayısıyla  $A^T P$  ve  $u^T P$  işlemlerinin gerçekleştirilebilmesi için RAM diskten yararlanılmıştır. RAM diskte depolanan  $A^T P$  ve  $u^T P$  matrisleri, sırasıyla  $23 \times 459 = 42$  Kb ve  $1 \times 459 = 2$  Kb yer kapladığından standart bellekte tutulabilir. Böylece  $A^T P A$ ,  $A^T P \ell$  ve  $u^T P u$  işlemleri standart bellek alanında daha hızlı yapılır. Kısaca, işlemlerin tümünde RAM disk kullanma zorunluluğu yoktur.

Yukarıda sözü edilen koşullar göz önünde bulundurularak gerçekleştirilen dengeleme işlemi yaklaşık 30 dakika sürmüştür. Ölçülerin dolu ağırlık matrisi  $P$ , bir deneysel kovaryans fonksiyonu yardımıyla türetilen ölçülerin kofaktörler matrisi  $Q$ 'nun inversi alınarak oluşturulmuştur. Matrisi parçalara ayırarak ve RAM disk kullanarak gerçekleştirilen invers alma işlemi de yaklaşık 30 dakikada tamamlanmıştır. Standart bellekte inversi alınan  $A_{22}$  ve  $B_{11}$  matrislerinin boyutları sırasıyla  $229 \times 229$  ve  $230 \times 230$  olarak belirlenmiştir. Bu matrislerin boyutunun büyüklüğü 640 Kb ile sınırlıdır ve tekil duyarlık kullanılarak yapılan denemeler standart bellekte inversi alınacak olan matrislerin boyutunun maksimum  $300 \times 300$  olabileceğini göstermektedir. Dolayısıyla inversi alınacak büyük matrisin boyutunun  $600 \times 600$ 'ü geçmemesi gerektiği söylenebilir. Diğer yandan, yayılmış belleğe doğrudan erişebilme olanağı bulunduğu da büyük matrisin

parçalara ayırma yöntemiyle alınmasında yarar vardır. Çünkü matrisin boyutu büyüdükçe normal bir algoritma ile invers almak için gereken süre katlanarak artmaktadır.

Yayılmış bellek kullanılarak gerçekleştirilen dengeleme işleminin, standart bellekte gerçekleştirilen ile süre bakımından karşılaştırmasını yapabilmek ve matrisin bir bütün halinde inversinin alınması ile parçalara ayrılarak inversinin alınması sonucu oluşan süre farkını vurgulamak amacıyla P matrisinin farklı boyutları kullanılarak yapılan denemeler Tablo-1'de sunulmaktadır.

Tablo-1: P Matrisinin farklı boyutları için standart ve yayılmış bellekte gerçekleştirilen işlemler

P Matrisinin Boyutu	İnvers Alma		Dengeleme Hesabı	
	Süre		Süre	
	Standart Bellek	Yayılmış Bellek	Standart Bellek	Yayılmış Bellek
100x100	2 Dak. 6 Sn.	1 Dak. 32 Sn.	8 Sn.	1 Dak. 9 Sn.
200x200	18 Dak. 45 Sn.	5 Dak. 48 Sn.	~ 28 Sn.	5 Dak. 13 Sn.
300x300	1 Saat 3 Dak.	17 Dak. 30 Sn.	Bellek Aşılıyor	12 Dak. 20 Sn.

Tablonun ikinci sütununda standart bellekte matrisin bir bütün halinde inversinin alınması sonucu boyut büyüdükçe sürenin katlanarak arttığı görülmektedir. Üçüncü sütunda ise yayılmış bellekte parçalara ayırarak invers alma işlemi için gerekli olan süreler sergilenmekte ve parçalara ayırma yöntemiyle alınan inversin süreyi oldukça kısalttığı görülmektedir. Bu yöntem standart bellek uygulandığında süre daha da kısalacaktır. Dört ve beşinci sütunlarda ise dengeleme işleminin standart ve yayılmış bellekte gerçekleştirilmesi sonucu harcanan sürelerin karşılaştırılması yapılmaktadır. Görüldüğü gibi yayılmış bellek kullanılarak gerçekleştirilen dengeleme işlemi daha uzun sürmektedir. Algoritma gereği standart bellek ile RAM disk arasında sürekli bir veri alışverişi vardır. Bu alışveriş rastgele erişimli dosyalar aracılığı ile gerçekleştirilmiştir. Rastgele erişimli dosyaların özelliği nedeniyle her bir matris verisi RAM diske bir kayıt numarası ile yazdırılmıştır. Matrislerin boyutu büyük olduğundan kayıt numaralarının uzun tamsayı (long integer) değişkenleri ile tanımlanması zorunluluğu doğmuş, bu değişkenler bellekte normal tamsayı değişkenlerine göre daha fazla yer kapladığından matris verilerine satır-sütun düzeninde erişmek için gerekli olan süre artmış ve bu da normal hesaplama süresini artırmıştır.

## 5. SONUÇ

DOS işletim sistemi ve DOS işletim sistemi altında çalışan uygulamalar en fazla 640 Kb'lık standart belleği kullanabilmektedir. Veri hacmi arttığında 640 Kb sınırı aşılır ve yayılmış belleği kullanma gereksinimi doğar. DOS, normalde 640 Kb sınırını aşamamasına karşın, kendi altında çalışan uygulama programlarının ürettiği ara değerleri RAM'in 1 Mb'dan sonraki bölümünde, yani yayılmış bellekte depolama olanağı sağlar. Yayılmış belleği doğrudan kullanabilmek için RAM disk oluşturmak gerekmektedir. RAM disk oluşturmak oldukça kolaydır; bunun için CONFIG.SYS dosyasında bir komut satırı tanımlamak yeterlidir. RAM disk, elektronik bir bellek olduğundan, bilgisayar kapatıldığında burada depolanan tüm veriler yok olur. Bu yüzden RAM diskte geçici depolanan bilgiler sabit diske aktarılmalıdır. Ayrıca RAM ne kadar büyüksse barındıracağı değer sayısı da o kadar fazla olacaktır.

Verilerin saklandığı dosyalar rastgele erişimli ya da ikili dosyalar olmalıdır. Çünkü sıralı erişimli (sequential) dosyaların aksine, bu tür dosyalarda herhangi bir veriye kolayca erişilebilir; ki matris verilerine satır ve sütun düzeninde ancak bu şekilde ulaşılmaktadır. Ayrıca bu dosyalarda saklanan veriler, sıralı erişimli dosyalarda saklananlara göre daha az yer kaplar (Nameroff, 1989).

Yayılmış bellek kullanılarak matris işlemlerinin gerçekleştirilmesinde matris boyutlarının büyüklüğü öncelikle yayılmış belleğin büyüklüğü ve oluşturulabilecek RAM diskin boyutu ile sınırlıdır. İnvers parçalara ayırma yöntemi ile alınacaksa, bu yöntemle göre standart bellekte inversi alınacak matrislerin boyutu maksimum boyutu belirlemede önem taşır. Bu çalışma kapsamında yapılan denemelere göre yayılmış bellekte inversi alınacak matrisin maksimum 600x600 boyutunda olması gerektiği söylenebilir.

Bu çalışmada, büyük matris işlemlerinin gerçekleştirilmesi için hazırlanan algoritmalarda, matris elemanları standart belleğe yalınlık açısından birer satır ya da sütun olarak taşınmışlardır. Satır ve sütun sayıları artırılarak optimum sonuçlara ulaşmak mümkündür. Son zamanlarda 80486 model kişisel bilgisayarlar oldukça yaygınlaşmaktadır. Bu model ile işlemler olabildiğince hızlı gerçekleştirilebilmektedir. Bu hız ile 640 Kb'ı aşan uygulamalarda RAM disk yardımıyla yayılmış bellekten yararlanma özelliği birleştirilirse, 80486 modeli daha etkin bir biçimde kullanılabilir. RAM diskin tamamen dolduğu durumlarda birkaç dosya için sabit disk de aynı amaçla kullanılabilir. Ya da işlem sırasında gereksiz dosyalar silinerek RAM diskte sürekli yer açılabilir.

**Teşekkür:** Çalışmamın oluşmasındaki değerli katkılarından dolayı Sayın Hocam Prof.Dr.Şerif HEKİMOĞLU'na ve bilgisayar konusundaki deneyimlerinden yararlandığım arkadaşım Arş.Gör.Gürsel GÜZEL'e teşekkür ederim.



## K A Y N A K L A R

- /1/ Ayres,F. : Matrices (Teori ve Problemlerle Matrisler),  
Türkçeye çeviren G.Oral, Sanem Çözümlü Serisi,  
Güven Kitabevi, Ankara, 1974.
- /2/ Jamsa,K. : DOS: The Complete Reference, Fourth Edition,  
Osborne Mc Graw-Hill, U.S.A., 1993.
- /3/ Korkmaz,H. : DOS, Halim Korkmaz, P.K.23, Cebeci / ANKARA, 1993.
- /4/ Mikhail,E.M., : Observations and Least Squares, IEP-Dun-Donnelley  
F.Ackermann. Harper & Row Publishers, 1976.
- /5/ Nameroff,S. : Quick Basic: The Complete Reference, Osborne Mc  
Graw-Hill, U.S.A., 1989.