



FATİH SULTAN MEHMET VAKIF ÜNİVERSİTESİ

Mühendislik Fakültesi

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Paralel Programlama Vize Raporu

BLOK BÖLÜTLENMİŞ MATRİS MATRİS ÇARPIMI

Furkan EREN

190221008

Dersin Danışmanı: Dr. Öğr. Üyesi Süha TUNA

İstanbul, Mayıs 2020

MATRİS BLOK BÖLÜTLEME

Matematikte, bir blok matris veya bölütlenmiş bir matris, bloklar veya alt matrisler olarak adlandırılan bölümlere ayrılmış olarak değerlendirilen bir matristir. Bir blok matris olarak değerlendirilen bir matris, onu parçalayan ya da daha küçük matrislerin bir parçası olarak ayıran yatay ve dikey çizgilerden oluşan bir orijinal matris parçasıdır. Herhangi bir matris, bir veya daha fazla yolla bir blok matris olarak ayrılabilir; her blok, satırlarının ve sütunlarının nasıl bölümlendiğiyle tanımlanır.

$$\begin{matrix} & A & & B & & C \\ \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix} & \times & \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix} & = & \begin{bmatrix} & & & \\ & & & \\ & & & \\ & & & \end{bmatrix} \end{matrix}$$

Resim 1: Matris Çarpımı

Resim 1' de A ve B matrislerinin bölütlenme işlemi yapılmadan önceki halleri gösterilmektedir. Bölütleme yapıldıktan sonraki A ve B matrislerinin görüntüsü ise Resim 2 'de bulunmaktadır.

$$\begin{matrix} & A & & B & & C \\ \begin{bmatrix} \begin{matrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{matrix} \\ \begin{matrix} 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{matrix} \end{bmatrix} & \times & \begin{bmatrix} \begin{matrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{matrix} \\ \begin{matrix} 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{matrix} \end{bmatrix} & = & \begin{bmatrix} & & & \\ & & & \\ & & & \\ & & & \end{bmatrix} \end{matrix}$$

Resim 2: Blok bölütlenmiş matris-1

A ve B matrisleri 1 yatay ve 1 dikey olmak üzere iki adet çizgi kullanılarak 4 parçaya ayrılmıştır. Her bir parça orijinal A ve B matrislerinin blok matrisleri olarak değerlendirilir.

$$\begin{matrix} & A & & B & & C \\ \begin{bmatrix} \begin{matrix} A1 & A2 \\ A3 & A4 \end{matrix} \end{bmatrix} & \times & \begin{bmatrix} \begin{matrix} B1 & B2 \\ B3 & B4 \end{matrix} \end{bmatrix} & = & \begin{bmatrix} & & & \\ & & & \\ & & & \\ & & & \end{bmatrix} \end{matrix}$$

Resim 3: Blok Bölütlenmiş matris-2

Resim 3' deki gibi bloklara ayrılmış yeni matrisimizin her bir blok parçasına yeni etiketler oluşturulmuştur. A matrisimizin blok matrisleri sırasıyla A1,A2,A3 ve A4 olmuştur. B matrisimizin blok matrisleri ise sırasıyla B1, B2, B3 ve B4 olarak adlandırılmıştır.

BLOK BÖLÜTLEME İLE MATRİS MATRİS ÇARPIMI

Blok bölütleme ile matris çarpımı işlemi orijinal bir matrisin bloklar haline dönüştürüldükten sonra blokların matris çarpımı kuralına göre birbirleriyle çarpılma işlemidir.

$$\begin{array}{ccc}
 \text{A} & & \text{B} \\
 \begin{bmatrix} \begin{array}{|c|c|} \hline 1 & 2 \\ \hline 5 & 6 \\ \hline 9 & 10 \\ \hline 13 & 14 \end{array} & \begin{array}{|c|c|} \hline 3 & 4 \\ \hline 7 & 8 \\ \hline 11 & 12 \\ \hline 15 & 16 \end{array} \\ \hline \end{bmatrix} & \times & \begin{bmatrix} \begin{array}{|c|c|} \hline 1 & 2 \\ \hline 5 & 6 \\ \hline 9 & 10 \\ \hline 13 & 14 \end{array} & \begin{array}{|c|c|} \hline 3 & 4 \\ \hline 7 & 8 \\ \hline 11 & 12 \\ \hline 15 & 16 \end{array} \\ \hline \end{bmatrix} = \begin{bmatrix} & \\ & \\ & \\ & \end{bmatrix} \\
 \\
 \begin{bmatrix} \begin{array}{|c|c|} \hline A1 & A2 \\ \hline A3 & A4 \\ \hline \end{array} \\ \hline \end{bmatrix} & \times & \begin{bmatrix} \begin{array}{|c|c|} \hline B1 & B2 \\ \hline B3 & B4 \\ \hline \end{array} \\ \hline \end{bmatrix} = \begin{bmatrix} & \\ & \end{bmatrix}
 \end{array}$$

Resim 4: Blok bölütlenmiş matris çarpımı-1

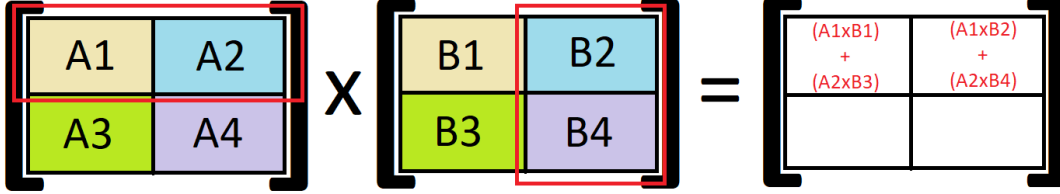
Orjinal A ve B matrisleri blok parçalara bölündükten sonra oluşan alt blok matrisler kendi aralarında matris çarpım kuralına göre çarpılır.

$$\begin{array}{ccc}
 \text{A} & & \text{B} \\
 \begin{bmatrix} \begin{array}{|c|c|} \hline A1 & A2 \\ \hline A3 & A4 \\ \hline \end{array} \\ \hline \end{bmatrix} & \times & \begin{bmatrix} \begin{array}{|c|c|} \hline B1 & B2 \\ \hline B3 & B4 \\ \hline \end{array} \\ \hline \end{bmatrix} = \begin{bmatrix} \begin{array}{|c|c|} \hline (A1 \times B1) & \\ \hline (A2 \times B3) & \\ \hline \end{array} \\ \hline \end{bmatrix}
 \end{array}$$

Resim 5: Blok bölütlenmiş matris çarpımı-2

Resim 5'te bloklar haline dönüştürülmüş A ve B matrislerinin ilk satır ve ilk sütun çarpımı gösterilmektedir. Bu işleme göre C matrisinin ilk bloğu şu şekilde hesaplanır;

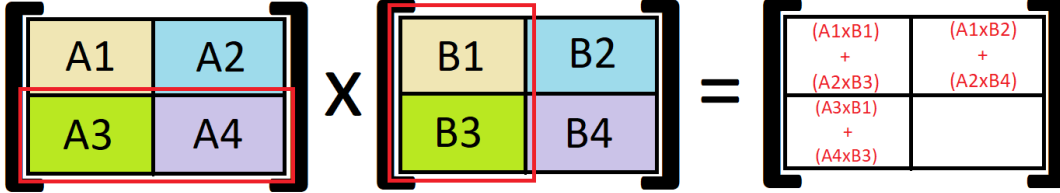
$$C1 = (A1 \times B1) + (A2 \times B3) \quad (1.1)$$



Resim 6: Blok bölütlenmiş matris çarpımı-3

Orjinal A matrisinin bloklara ayrılmış ilk satırı ile B matrisinin bloklar halinde düzenlenmiş ikinci sütunu çarpılarak C matrisinin 2. bloğu elde edilir.

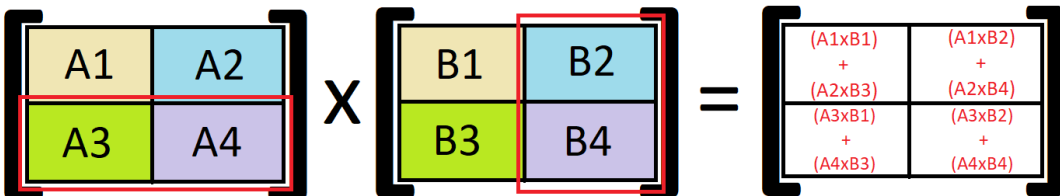
$$C2 = (A1 \times B2) + (A2 \times B4) \quad (1.2)$$



Resim 7: Blok bölütlenmiş matris çarpımı-4

Orjinal A matrisinin bloklar olarak ayrılmış olan 2. satırı ile B matrisinin bloklar olarak düzenlenmiş ilk sütunu çarpıldığında C matrisinin 3. bloğu elde edilir.

$$C3 = (A3 \times B1) + (A4 \times B3) \quad (1.3)$$



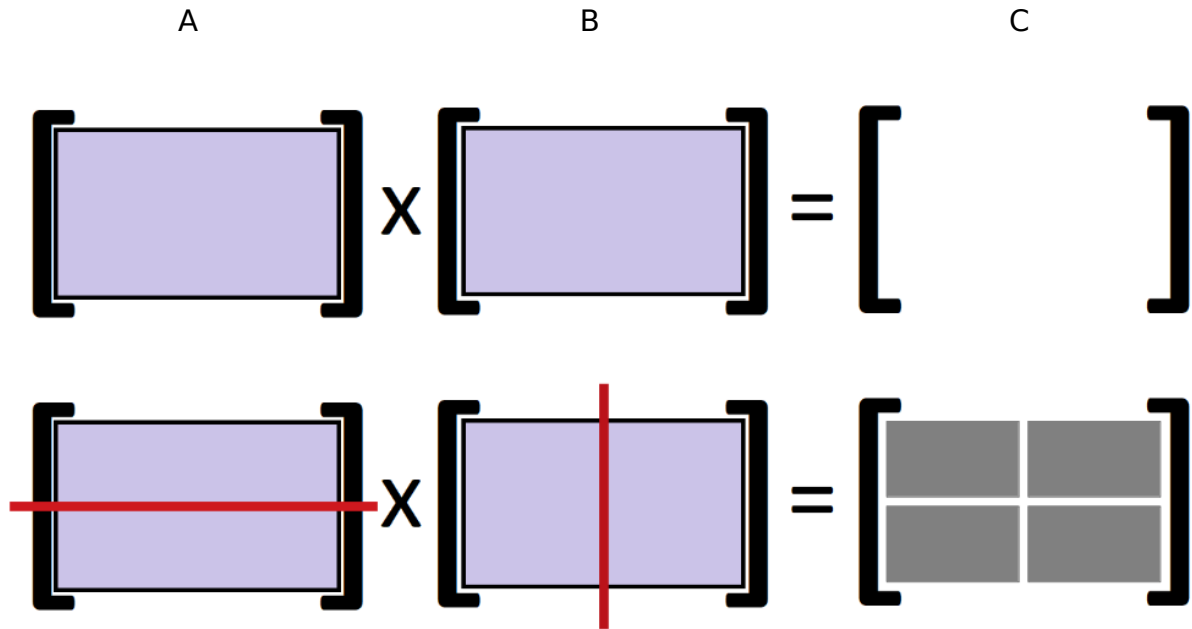
Resim 8: Blok bölütlenmiş matris çarpımı-5

Son olarak orijinal bloklara ayrılmış A matrisinin 2. satırı ve B matrisinin 2. sütunu çarpıldığı zaman C matrisinin 4. son bloğu elde edilmiş olur.

$$C_4 = (A_3 \times B_2) + (A_4 \times B_4) \quad (1.4)$$

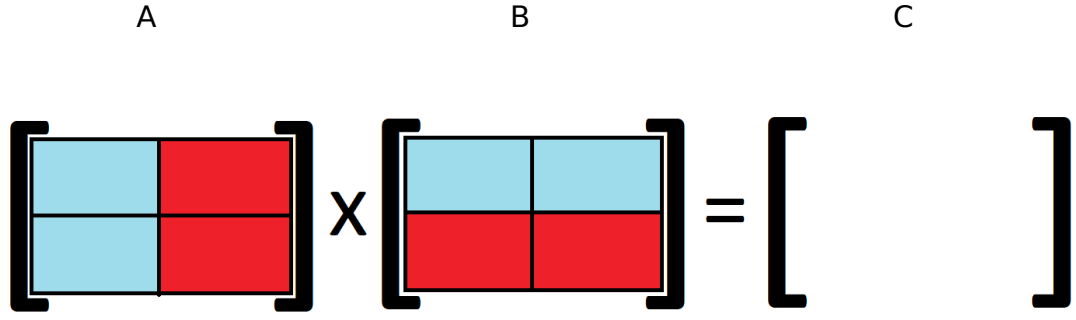
BLOK BÖLÜTLENMİŞ MATRİS ÇARPIMININ PARALEL PROGRAMLAMA İLE UYGULANMASI

Blok bölütlenmiş matris çarpımının paralel programlama ile hesaplaması için bu çalışmada OpenMpı kullanılmıştır. Matrislerin bloklar halinde paralel olarak dağıtılabilmesi için A ve B matrisleri ilk olarak ikiye bölünmüştür. A matrisi yatay yönde bölünürken B matrisi dikey yönde 2'ye ayrılmıştır. Bu ayrıştırma işleminin asıl amacı A matrisinin sütun vektörleri cinsinden paralel olarak dağıtılabilmesini sağlamaktır. B matrisinin ise dikey olarak 2'ye bölünmesinin sebebi B matrisinin satır vektörleri şeklinde paralel olarak dağıtılabilmesini sağlamak.



Resim 9: Matrisleri 2 parçaya ayırma işlemi

A ve B matrisleri sırasıyla yatay ve dikey olarak ikiye ayrıldıktan sonra blok parçalar halinde bölütleme için hazır hale getirilmiş olur. Yatay olarak ayrılmış A matrisinin 1. satırı ve 2. satırı, sütun vektörleri şeklinde paralel olarak dağıtılır. Dikey olarak ikiye ayrılmış B matrisinin 1. ve 2. sütunları satır vektörleri şeklinde paralel olarak dağıtılır.

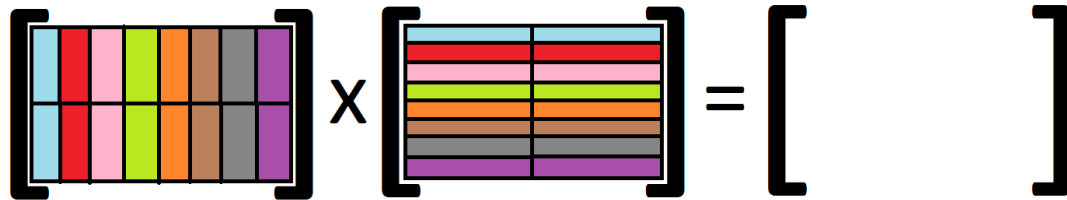


Resim 10: 2 işlemciye paralel olarak dağıtılmış A ve B matrisleri

Resim 10' da yatay olarak ayrılmış A matrisinin ilk satırı sütun vektörleri şeklinde 2 işlemciye paralel olarak dağıtılmıştır. Aynı şekilde A matrisinin ikinci satırı da sütun vektörleri şeklinde 2 işlemciye paralel olarak dağıtılmıştır. Dikey olarak ikiye ayrılmış olan B matrisinin ilk sütunu satır vektörleri şeklinde 2 işlemciye paralel olarak dağıtılmıştır. B matrisinin ikinci sütunu da aynı yöntemle satır vektörleri cinsinden 2 işlemci için paralel olarak dağıtılmıştır.



Resim 11: 4 işlemci için paralel olarak dağıtılan A ve B matrisleri



Resim 12: 8 işlemci için paralel dağıtılmış A ve B matrisleri

Blok bölütlenmiş şekilde paralel olarak dağıtılan A ve B matrisleri yerel işlemcilerde matris çarpım kurallarına göre çarpılır. Çarpım sonucunda 'Master' işlemcide sonuçlar toplanarak blok bölütlenmiş matris matris çarpımı işlemi hesaplanır.

SONUÇLAR

Blok bölütlenmiş matris matris çarpımı için OpenMpi ile paralel olarak programlanmış C programla dili ile işlemci sayısına göre analizler yapılmıştır.

Analizi gerçekleştirilen bilgisayarın donanım özellikleri ise şu şekildedir;

İşlemci = İ7-4710HQ(4 çekirdek,8 iş parçaçığı)

Ram =16 gb DDR4

Depolama =128 gb SSD

Blok bölütlenmiş matris çarpımı 3 farklı işlemci sayısına göre test edilmiştir. 1, 2, ve 4 adet işlemci ile 4 farklı matris boyutu çarpılmıştır.

Tablo1: Analiz edilen A ve B matrisi boyutları

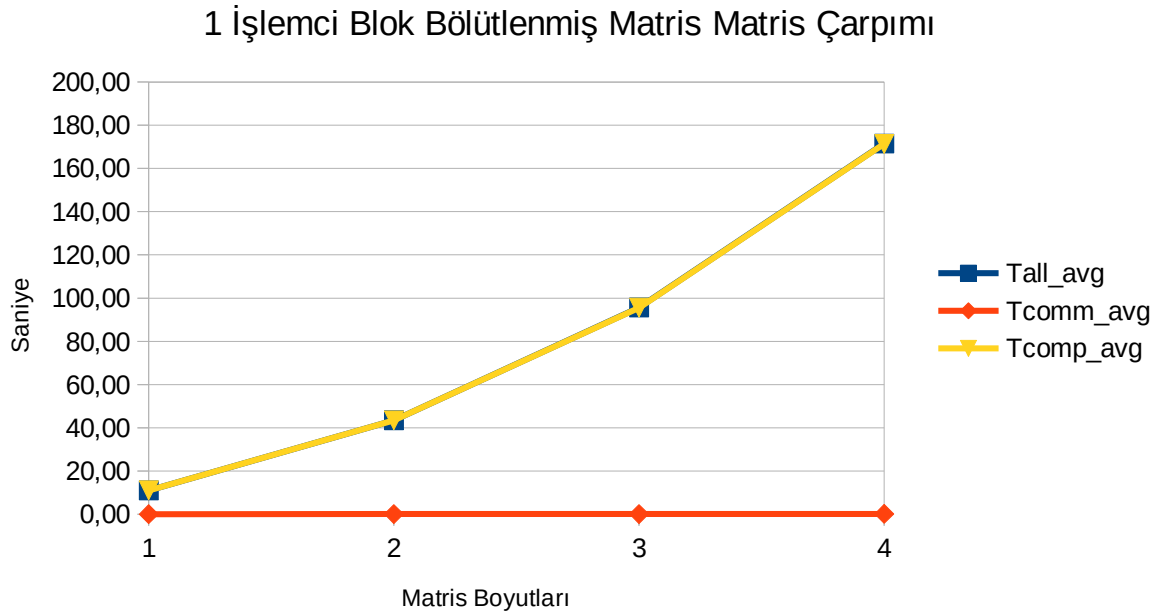
A Matrisi	B Matrisi
2000x2000	2000x1000
4000x4000	4000x1000
6000x6000	6000x1000
8000x8000	8000x1000

1 İŞLEMCI

1 işlemci ile 4 farklı matris boyutu için blok bölütlenmiş matris çarpımı uygulanmıştır. Matris çarpımı için 3 farklı sonuç analiz edilmiştir. Çarpım için geçen toplam süre, işlemciler arası iletişim için geçen toplam süre ve son olarak hesaplama kısmı için geçen toplam süreler hesaplanmıştır.

Tablo 2: 1 İşlemci için blok bölütlenmiş matris çarpımı süre karşılaştırmaları

1.Matris	2.Matris	Processes	Tall_avg	Tcomm_avg	Tcomp_avg
2000x2000	2000x1000	1	10,997008	0,015020	10,981987
4000x4000	4000x1000	1	43,465123	0,043344	43,421779
6000x6000	6000x1000	1	95,632902	0,084028	95,548875
8000x8000	8000x1000	1	171,502802	0,140030	171,362772



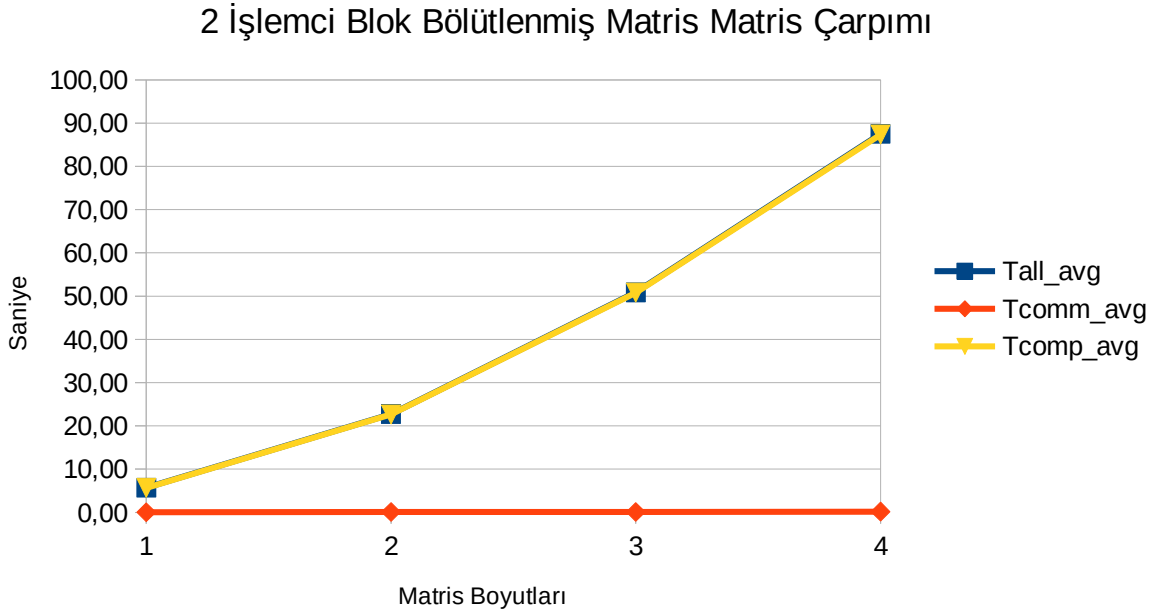
Resim 13: 1 işlemci ile blok bölütlenmiş matris çarpımı grafiği

2 İŞLEMCI

2 işlemci ile 4 farklı matris boyutu için blok bölütlenmiş matris çarpımı uygulanmıştır. Matris çarpımı için 3 farklı sonuç analiz edilmiştir. Çarpım için geçen toplam süre, işlemciler arası iletişim için geçen toplam süre ve son olarak hesaplama kısmı için geçen toplam süreler hesaplanmıştır.

Tablo 3: 2 işlemci için blok bölütlenmiş matris çarpımı süre karşılaştırmaları

1.Matris	2.Matris	Processes	Tall_avg	Tcomm_avg	Tcomp_avg
2000x2000	2000x1000	2	5,699490	0,029456	5,670034
4000x4000	4000x1000	2	22,734847	0,092479	22,642368
6000x6000	6000x1000	2	50,879290	0,090444	50,788846
8000x8000	8000x1000	2	87,511726	0,169013	87,342713



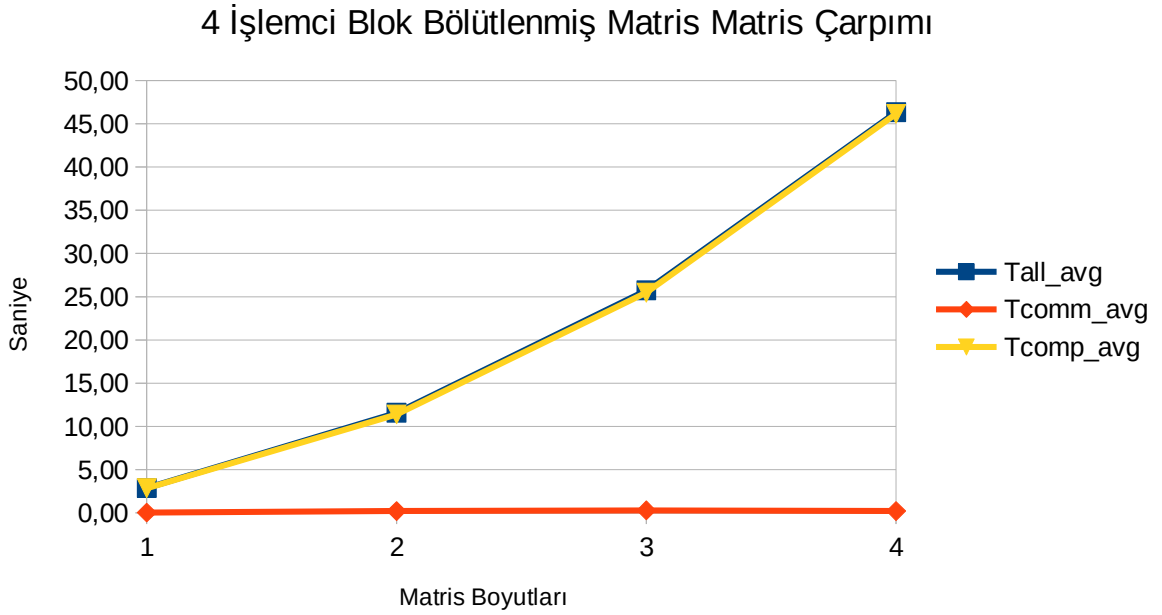
Resim 14: 2 işlemci ile blok bölütlenmiş matris çarpımı grafiği

4 İŞLEMCI

4 işlemci ile 4 farklı matris boyutu için blok bölütlenmiş matris çarpımı uygulanmıştır. Matris çarpımı için 3 farklı sonuç analiz edilmiştir. Çarpım için geçen toplam süre, işlemciler arası iletişim için geçen toplam süre ve son olarak hesaplama kısmı için geçen toplam süreler hesaplanmıştır.

Tablo 4: 4 işlemci için blok bölütlenmiş matris çarpımı süreleri

1.Matris	2.Matris	Processes	Tall_avg	Tcomm_avg	Tcomp_avg
2000x2000	2000x1000	4	2,858957	0,021054	2,837903
4000x4000	4000x1000	4	11,572647	0,193767	11,378880
6000x6000	6000x1000	4	25,729125	0,271678	25,457447
8000x8000	8000x1000	4	46,334633	0,194066	46,140567



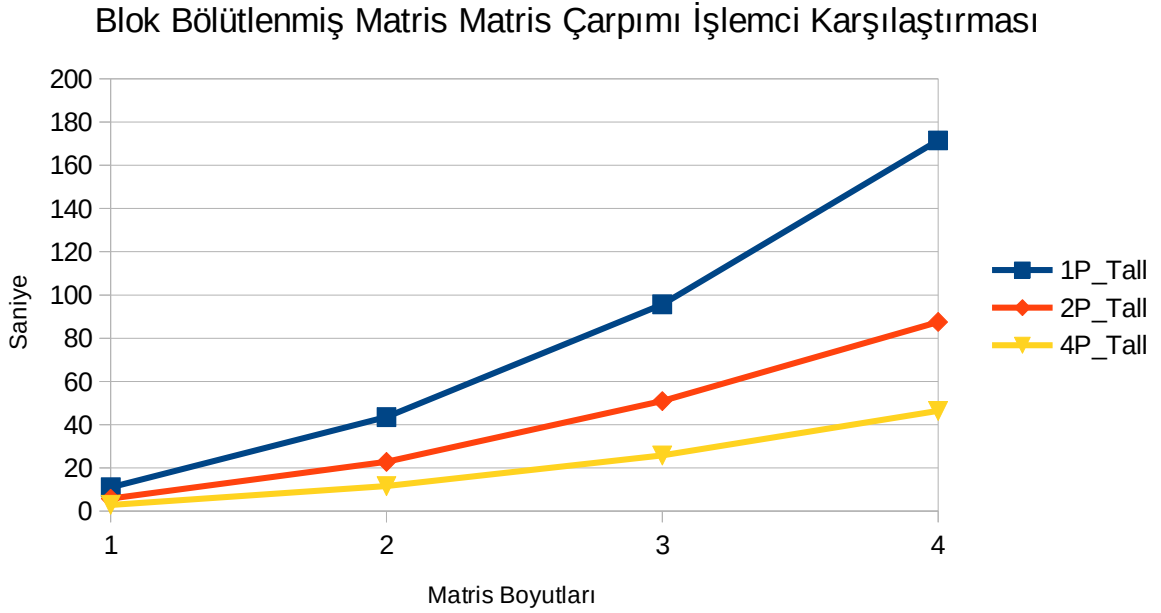
Resim 15: 4 işlemci için blok bölütlenmiş matris çarpımı grafiği

KARŞILAŞTIRMA

Blok bölütlenmiş matris çarpımı için 3 farklı işlemci sayısı ve 4 farklı matris boyutu kullanılmıştır. 3 farklı işlemci sayısı için farklı matris boyutlarına göre blok bölütlenmiş matris çarpımı için geçen toplam süreler tablo 5 ve resim 16' da karşılaştırılmıştır.

Tablo 5: İşlemci sayılarına göre blok bölütlenmiş matris çarpımı toplam süreleri

1.Matris	2.Matris	1P_Tall	2P_Tall	4P_Tall
2000x2000	2000x1000	10,997008	5,69949	2,858957
4000x4000	4000x1000	43,465123	22,734847	11,572647
6000x6000	6000x1000	95,632902	50,87929	25,729125
8000x8000	8000x1000	171,502802	87,511726	46,334633



Resim 16: Farklı işlemci sayıları için blok bölütlenmiş matris çarpım süreleri grafiği

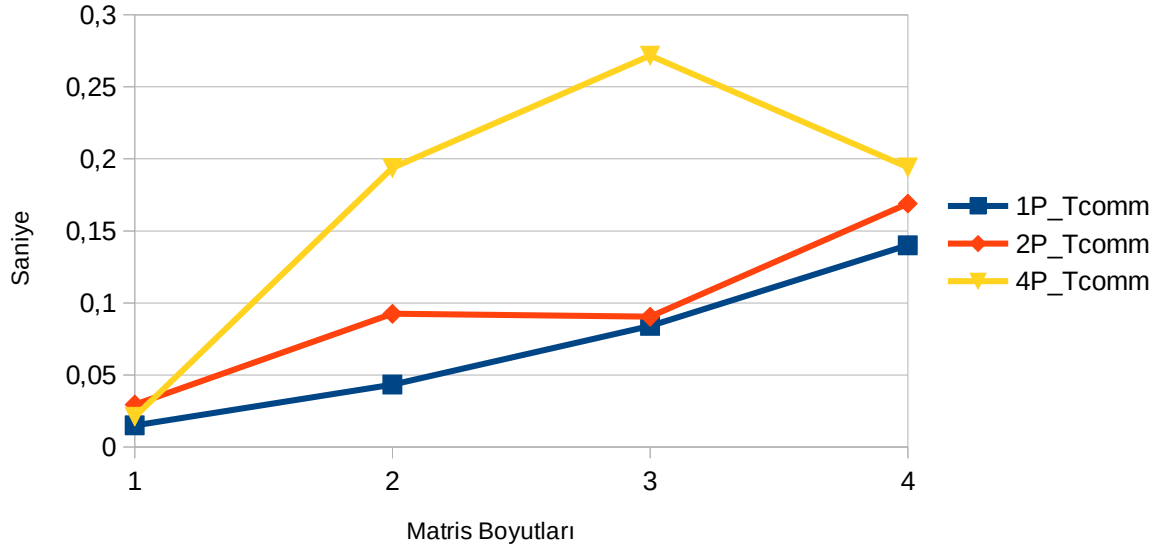
Tablo 5 ve Resim 16'daki grafik incelendiğinde işlemci sayısı arttıkça toplam hesaplanma sürelerinde ciddi azalmalar meydana gelmektedir. Yaklaşık olarak ilk 4 işlemci için işlemci sayısı 2 katına çıktıkça hesaplanan süre yarıya düşmektedir. Matris boyutları arttıkça toplam süreler arasındaki fark gözle görülür bir şekilde değişmektedir.

İşlemci sayısının artması paralel programlamada toplam süreyi ciddi derecede azaltmaktadır. İşlemci sayısının artması blok bölütlenmiş matrislerin işlemcilere dağıtılmasında arttırmaktadır. İşlemci sayısına ve matris boyutuna göre işlemciler arası iletişim için geçen toplam süreler tablo 6 ve resim 17' de karşılaştırılmıştır.

Tablo 6: İşlemci sayısına göre blok bölütlenmiş matris çarpımında işlemciler arası iletişim için geçen toplam süre karşılaştırması

1.Matris	2.Matris	1P_Tcomm	2P_Tcomm	4P_Tcomm
2000x2000	2000x1000	0,01502	0,029456	0,021054
4000x4000	4000x1000	0,043344	0,092479	0,193767
6000x6000	6000x1000	0,084028	0,090444	0,271678
8000x8000	8000x1000	0,14003	0,169013	0,194066

Blok Bölütlenmiş Matris Çarpımı İşlemci İletişimi Süre Karşılaştırması



Resim 17: Blok bölütlenmiş matris çarpımı işlemci iletişimi süre karşılaştırması grafiği

Grafik incelendiğinde işlemci sayısına göre toplam geçen süreler karşılaştırıldığında en az süre 4 işlemci için elde edilmişken işlemciler arası iletişimde geçen toplam sürede ise 4 işlemci ile yapılan hesaplama en fazla iletişim süresini oluşturmuştur. İşlemci sayısı arttıkça iletişim için gereken süre miktarı artmaktadır fakat toplam süreye olan azaltıcı etkisi ile karşılaştırıldığında ihmal edilebilir seviyelerde olduğu gözlemlenmektedir.