Paralel Hesaplama ve MATLAB Uygulamaları

Muammer AKÇAY [1] , Hüseyin Aşkın ERDEM [2]

[1] Dumlupınar Üniversitesi, Bilgisayar Mühendisliği Bölümü, Kütahya, Türkiye

[2] Dumlupınar Üniversitesi, Bilgisayar Mühendisliği Bölümü, Kütahya, Türkiye

makcay@dpu.edu.tr, haerdem@mail.dpu.edu.tr

Özet: Bu çalışmada temel olarak paralel hesaplama konusu üzerinde çalışılmıştır. Paralel hesaplamanın avantajları ve dezavantajları açıklanmıştır. Paralel hesaplama eğitiminde kullanılabilecek programlar ve bu programlara ait örneklere yer verilmiştir. Ayrıca paralel hesaplama kütüphaneleri, paralel hesaplama sistemleri, paralel hesaplamada performans ve uygulama gibi konular da incelenmiştir.

Anahtar Sözcükler: Paralel Hesaplama, Paralel Hesaplamada Kütüphaneler, Paralel Hesaplamada Performans, Paralel Hesaplamada Uygulamalar.

Parallel Computing And MATLAB Applications

Abstract: In this study, parallel computing basically examined. Advantages and disadvantages of parallel computing are explained. Programs that can use for parallel computing education and examples about these programs are placed in this article. Also in this work, parallel computing libraries, parallel computing systems, performance and applications are examined.

Keywords: Parallel Computing, Parallel Computing Libraries, Parallel Computing Performance, Parallel Computing Applications.

1. Giriş

Paralel hesaplamada yapılacak bir iş, birden çok iş bölümlerine ayrılır ve her iş bölümünde bulunan sonuçlar bir yerde toplanır (örneğin, ana bilgisayar) ve bu şekilde, bu işlerin daha kısa zamanda ve daha hızlı bir şekilde yapılmasına olanak sağlanır.[1]

Günümüzde, artık çoğu bilgisayarda çift çekirdekli işlemcilerin (hatta dört çekirdeklileri de piyasaya çıkmıştır) kullanılmasıyla paralel hesaplamayla yapılabilecek projelere yeni alanlar açılmış ve bu sayede paralel hesaplama üzerindeki ilgi daha da artmıştır.

Paralel hesaplama ile sonuca daha hızlı ulaşmanın yanında iş bölümleri aynı anda işletilerek eş zamanlı olarak da işlemlerin yapılmasıyla performans da arttırılır.

Çift ve çok çekirdekli işlemciler ile artık, bilgisayarların, işlemleri daha hızlı ve kısa sürede yapabilmeleri, hatta daha az ısınmaları da sağlanabilmektedir.

Paralel hesaplamaları gerçekleştirmek için ya istemci-sunucu (client-server) tarzı bir yaklaşım ya da Paralel Hesaplama için geliştirilmiş özel yöntem ve programlar, örneğin Mpich2 [11] kullanılabilir.

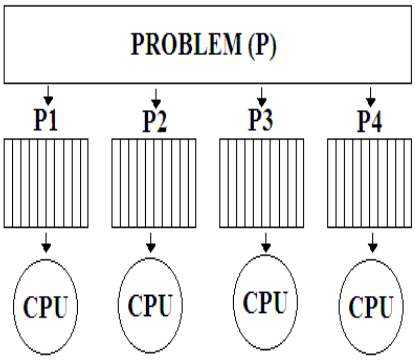
Bir sonraki bölümde paralel hesaplamanın neden kullanılması gerektiğine ve ne zaman kullanılması gerektiğine ilişkin bilgiler verildi.

2.1 Neden Paralel Hesaplama?

Aşağıdakilerden dolayı paralel hesaplama kullanılabilir:
-Bellek kullanımını en uygun şekilde gerçekleştirerek bilgisayardaki yavaşlamaları engellemesi,
-Tek CPU'nun yerine çoklu (çift,dört, vs.) CPU kullanılması ile sıcaklık artışının engellenmesi,
-Hesaplamaların hızlı bir şekilde yapılmasını sağlama.
-Hızlı hesaplama ile performans artışı.

Paralel hesaplamayla **bir problemin;**

1)Birden çok parçaya ayrılması sağlanır.
2)Bu parçalar eşit zaman aralıklarına bölünür.
3)Her bir iş bölümü farklı CPU'larda aynı zamanda eş zamanlı olarak işleme alınır, hesaplama yapılır.



Şekil 1. Bir problemin eş parçalara ayrılarak eş zamanlı olarak farklı CPU'larda çalıştırılması [3].

Sonraki bölümde paralel hesaplamanın tarihçesi incelendi.

2.2 Paralel Hesaplama Tarihçesi

Paralellik ile ilgili ilk fikirler, ilk olarak 1958'de IBM [27] araştırmalarında nümerik hesaplamaların yapılabileceği fikrini ortaya atan John Cocke ve Daniel Slotnick ile ortaya çıkmıştır.
1960'a gelindiğinde Novosibirsk Matematik Enstütüsünden (IMN) E. V. Yevreinov'un birbirine bağlantılı şekilde programlanabilen paralel mimarileri tasarlamasıyla paralel kavramı yeni bir boyut kazandı.
1964'de ise Daniel Slotnick Lawrence Livermore National Laboratuarlarında kullanılmak üzere büyük-paralel makineleri geliştirdi.
1967'de Gene Amdahl ve Daniel Slotnick AFIPS Konferansında paralel işlemlerin olasılığıyla ilgili bir makale yayınladılar. Bu paralellik ile ilgili olan tartışma konusu daha sonra "Amdahl Kanunu" [28] olarak adlandırıldı.
1983'te Goodyear Aerospace NASA Goddard için Massively Paralel Processor (MPP) [29] geliştirdi.
1985'te David Gelernter Linda paralel programlama sisteminin temellerini ortaya koydu.
1986'da Parallel Virtual Machine (PVM) [18] projesi dağıtık bilgisayarlarda gerekli olan yazılımların kullanılabilmesi için geliştirildi.
1993'te IBM, RISC RS/6000 işlemcisine dayalı ilk SP1 Powerparallel sistemi piyasaya sürdü [5, 25].

Bir sonraki bölümde paralel hesaplamadaki avantaj ve dezavantajlardan bahsedildi.

2.3 Paralel Hesaplama Avantajları

Paralel hesaplamanın en önemli avantajlarından birisi problemin çözüm zamanını azaltarak çözüme daha hızlı bir şekilde ulaşılmasını sağlamaktır. Problem bir parça olarak değil parçalar halinde ve her bir parçanın belirli zaman aralıklarına bölünmüş olması sayesinde daha kolay ve hızlı çözüm elde edilmesi sağlanır [3].

2.4 Paralel Hesaplama Dezavantajları

Paralel hesaplamada parçalara ayrılan problemlerin her biri ayrı bilgisayarlarda veya CPU'larda çalıştırıldığı için bu parçaların herhangi birinde oluşabilecek bir çözümsüzlük veya hata diğer parçalar tamamen çözülse bile problemin genel olarak başarıya ulaşmasına engel olabilir. Ayrıca bu parçaların her birine zaman kısıtı konulduğu için her bir görev kendine verilen süre zarfı içerisinde çözülmüş olmalıdır. Çözülecek veya çözülmüş parçalar birbirleriyle etkileşimde olmalı ve sürekli olarak iletişimlerini korumalıdırlar [13].

Bir sonraki bölümde paralel hesaplamada kullanılan kütüphaneler ele alındı.

3.Paralel Hesaplama Kütüphaneleri

Paralel hesaplama eğer programla yapılacaksa bunun için hazırlanmış MPI (Message Passing Interface) Kütüphanesi [16] kullanılabilir. MPICH[15] ve LAM[16] uygulamaları da, MPI ile kullanılabilir [4].

Diğer paralel hesaplama kütüphaneleri:
Co-Array Fortran [17], Parallel Virtual Machine [18], Global Arrays [19], OpenMP[20] , High Performance Fortran HPF [21] , Unified Parallel C/UPC [22] , SHMEM [23], Occam[8], Cilk[12].[1]

Bundan sonraki bölümde paralel hesaplamayla ilgili uygulamalar ele alındı.

4. Paralel Hesaplama Uygulamaları

4.1 Paralel Hesaplama Gerçeklenmesi

Paralel hesaplamayı yani bir programı birden çok parçaya ayırarak eş zamanlı olarak çözme işlemini uygulamak amacıyla bir örnek uygulama gerçekleştirildi.

Bunun için birkaç farklı program kullanılarak üç (3) değişik yolla çözüm üretilebilir.

1.)Kullanılan işletim sistemi üzerine sanal makine (virtual box [8]) kurularak, yine bu sanal makine içerisine bir veya birden fazla işletim sistemi yüklenebilir. Bu şekilde çözümü yapılacak problem (program) bu virtual box içindeki işletim sistemlerine paylaştırılabilir.

2.)Matlab [9] gibi programlar kullanılarak, çözüm gerçekleştirilebilir. Bunun için Matlab'ın Paralel Toolbox altyapısından yararlanılabilir.

3.)Microsoft Visual Studio 6.0 Enterprise Edition [10] ile bağlantılı çalışan Mpich2 [11] programı ile kullanılan MPI kütüphanesi [16] kullanılarak paralel hesaplama işlemi gerçeklenebilir.

4.2 Paralel Hesaplama Örneği

MPI kütüphanesi ile kodlanmış program örneği: [6]

Örnekte MPI kütüphanesinden faydalanılarak iki nokta arasındaki iletişim sağlanır.

```
#include <stdio.h>
#include <mpi.h>

int main(int argc, char **argv)
{
   int id, numprocs;
   MPI_Init(&argc,&argv);
   MPI_Comm_size(MPI_COMM_WORLD,
&numprocs);
```

```
MPI_Comm_rank(MPI_COMM_WORLD, &id);
   if (id==0)
   printf("Merhaba! Ben islemci 0. Toplam %d islemci var.\n", numprocs);
   MPI_Finalize();
   return 0;
}
```

4.3 MATLAB ile Paralel Hesaplama

Paralel hesaplamada yapılacak bir iş, birden çok iş bölümlerine ayrılıp, iş bölümleri, farklı işlemciler veya bilgisayarlar üzerinde işlenerek ve her iş bölümünde bulunan sonuçların bir yerde toplanıp (örneğin, ana bilgisayar) ve bu şekilde, bu işlerin daha **kısa zamanda** ve daha **hızlı** bir şekilde yapılmasına olanak sağlamaktadır. Bu yapılacak görevin ayrıldığı iş bölümlerinin farklı bilgisayarlara dağıtılması Matlab-Parallel Computing ToolBox [14] ile gerçeklenebilir.
Şimdi Matlab-Parallel Toolbox [14] kullanarak nasıl paralel hesaplama yapılabileceğini görelim:

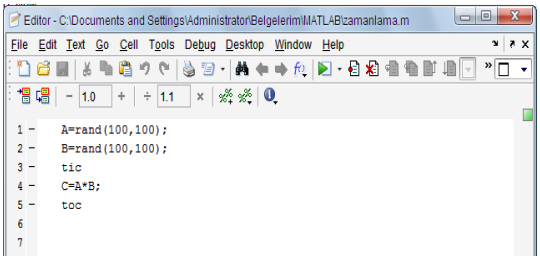
MATLAB [9] ve Simulink yazılım ortamı, paralel hesaplama araçlarıyla hem çalıştığınız bilgisayarda hem de başka bir bilgisayar kümesinde (çok çekirdekli bilgisayar) hesapları dağıtabilmektedir.

Matlab Paralel Toolbox'a [14] Giriş:

Bir örnek ile performans analizi yapıldı:

Önce Matlab Paralel Toolbox kullanmadan bir "m-file" oluşturularak 100X100'lük iki matrisi (A ve B matrisi) rastgele oluşturan ve bu iki matrisin çarpımını hesaplayan program command window ile paralelleme özelliğini kullanmadan çalıştırıldı.
Yalnız programımıza hesaplamanın ne kadar zaman alacağını görebilmek için çarpımın gerçekleştirilebileceği satırın öncesi ve sonrasındaki satırlara tic-toc adımları eklendi.

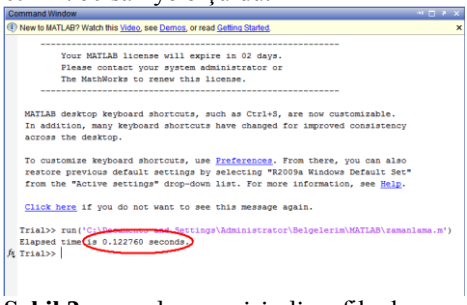
Kullanacağımız m-file (zamanlama.m) dosyası aşağıdaki gibi kodlandı:



Şekil 2.zamanlama.m isimli m-file dosyası.

```
A=rand(100,100);
B=rand(100,100);
tic
C=A*B;
toc
```

Oluşturulan m-file dosyası Command Window'da çalıştırıldığında programın toplam çalışma zamanı "elapsed time" olarak 0.122760 saniye ölçüldü.

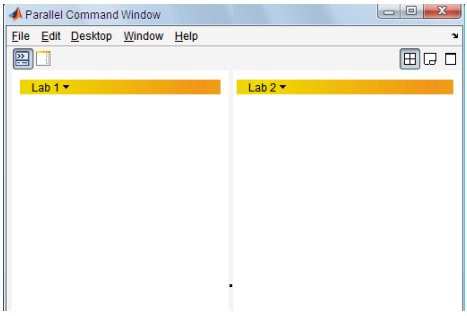


Şekil 3 zamanlama.m isimli m-file dosyasının Command Window'da hesaplanma süresi.

Bu çarpım işlemi daha sonra Matlab Paralel Toolbox [14] kullanılarak gerçeklendi [24].

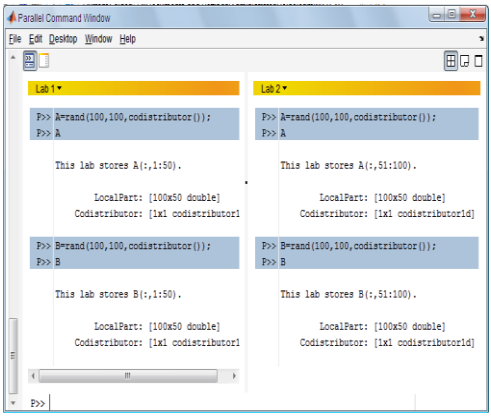
Matlab Paralel Toolbox [14]'ı kullanabilmek için önce
>> **pmode start local 2**
komutu ile, **paralel command window** (paralel toolbox ile kullanılabilen iş bölümlerinin (lab)) açıldı. Burada lab'lar arası veri alış-verişi sağlanır. Eğer veriler çok büyük ve karmaşık işlemler gerektiriyorsa, bu veriler farklı işlem birimlerine (lab) dağıtılır.

Her birim diğerleriyle bağlantılı çalışmalıdır, yani birbiriyle sürekli iletişim içinde olmalıdırlar.



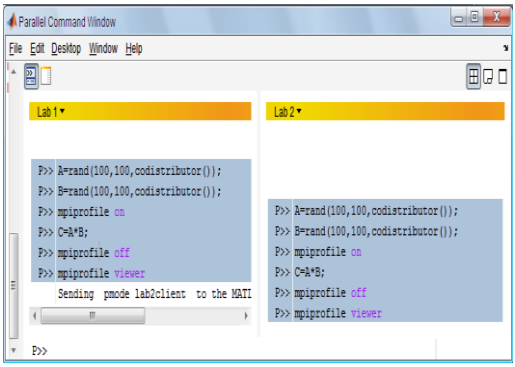
Şekil 4. Parallel Command Window [14] ile oluşturulmuş iki adet çalışma labı.

Oluşturulan bu iki çalışma lab'ına iki matrisi (A ve B) ayrıldı;
Bunun için bu iki matris aşağıdaki şekilde tanımlanıp lab'lara dağıtıldı:
P>>A=rand(100,100,codistrubutor());
P>>B=rand(100,100,codistrubutor());



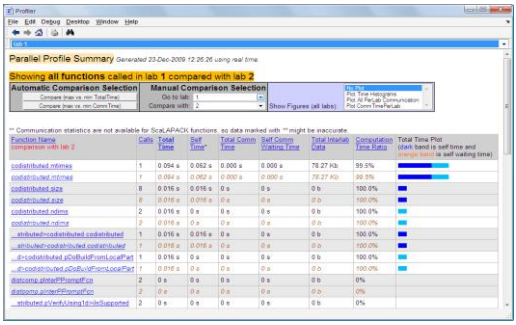
Şekil 5. Parallel Command Window [14] ile iki matrisin dağıtılması.

Matris çarpım işleminin ne kadar zaman alacağını görebilmek için çarpım komut satırı **mpiprofile on** ve **mpiprofile off** komutuyla kodlandı. Son olarak **mpiprofile viewer** komutuyla profiler penceresi açılarak programın performansı gözlendi.

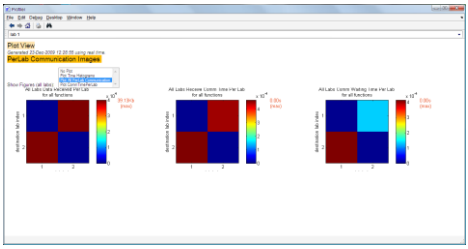


Şekil 6. Parallel Command Window [14] ile mpiprofile kullanımı.

Açılan profiler ekranında lab'lara ait zaman değerleri hesaplandı. Bu değerler istenirse karşılaştırmalı olarak da, yani hem lab1 hem de lab2 olarak ayrı ayrı incelenebilir.



Şekil 7. Profiler ile hesaplama süreleri grafiği.



Şekil 8. Profiler ile görsel grafikler.

Paralel Command Window penceresi **P>>pmode exit** komutu ile kapatıldı.

Kısaca; Command Window ile 0.122760 saniyede gerçekleştirilen hesaplama örneği, Paralel Command Window (Matlab Parallel ToolBox) kullanarak aynı işlem 0.094 saniyede gerçekleştirildi [24].

5. Sonuç ve Öneriler

Bu çalışmada paralel hesaplama konusu üzerinde çalışıldı. Paralel hesaplamada amaç, problemlerin küçük iş bölümlerine ayrılması ve bunların aynı zaman aralıklarında hesaplanıp, sonucun bir noktada (ana bilgisayar) birleştirilerek işlemlerin daha kısa sürede, daha hızlı ve daha etkin bir şekilde yapılabilmesi sağlanır. Paralel hesaplama programla gerçeklenmek isteniyorsa çeşitli kütüphaneler (MPI gibi) veya Matlab Paralel ToolBox [14] kullanılarak paralel hesaplamaları ve işlemleri gerçekleştirilebilir.

5. Kaynaklar

[1] Paralel Hesaplama hakkında ansiklopedik bilgi:
http://www.turkcebilgi.com/paralel_hesaplama/ansiklopedi

[2] Paralel Hesaplama nedir?:
http://www.programlamarehberi.com/sorucevap-5-nedir_bu_.html

[3] MPI Programlamaya giriş ve motivasyon (Kısım 1):
http://www.uybhm.itu.edu.tr/documents/basarim09sunum/01_Giris_ve_Motivasyon_akinci_v3.pdf

[4]MPICH2 kullanarak Windows,LINUX veya “Windows veya LINUX” İşletim Sistemleri:
http://www.emresururi.com/mpich2/mpich2_rehberi_v1_2.pdf

[5] The History of the Development of Parallel Computing:
http://ei.cs.vt.edu/~history/Parallel.html

[6] MPI Örnekleri:
http://www.grid.org.tr/etkinlikler/egitim/sunumlar/mpi_ornekleri.pdf

[7] Introduction To The Programming Language Occam:
http://www.cs.otago.ac.nz/cosc461/occam.pdf

[8] Virtual Box Resmi Sitesi:
www.virtualbox.org/

[9] MATLAB Resmi Sitesi:
http://www.mathworks.com/

[10] Microsoft Visual Studio 6.0 Enterprise Edition Resmi Sitesi:
http://msdn.microsoft.com/en-us/library/ms950417.aspx

[11] MPICH 2 Resmi Sitesi:
http://www.mcs.anl.gov/research/projects/mpich2/

[12] Multithreaded Programming In Cilk:
http://www.ll.mit.edu/HPEC/agendas/proc07/Day3/25_Frigo_Pres.pdf

[13]Cluster Computing – Fırat Üniversitesi (Galip AYDIN):
http://web.firat.edu.tr/bilmuh/gaydin/teaching.html#

[14] MATLAB Paralel Computing Toolbox 4.2 Resmi Sitesi:
http://www.mathworks.com/products/parallel-computing/

[15] MPICH Resmi Sitesi:
http://www.mcs.anl.gov/research/projects/mpi/mpich1/

[16]LAM/MPI Parallel Computing Resmi Sitesi:
http://www.lam-mpi.org/

[17] Co-Array FORTRAN Resmi Sitesi:
http://www.co-array.org/

[18] PVM-Parallel Virtual Machine Resmi Sitesi:
http://www.csm.ornl.gov/pvm/

[19] Global Arrays Resmi Sitesi:
http://www.emsl.pnl.gov/docs/global/

[20] OpenMP Resmi Sitesi:
http://openmp.org/wp/

[21] High Performance Fortran (HPF) Resmi Sitesi:
http://www.netlib.org/hpf/

[22] Unified Parallel C (UPC) Resmi Sitesi:
http://upc.lbl.gov/

[23] SHMEM Programming Manual:
http://staff.psc.edu/oneal/compaq/ShmemMan.pdf

[24]Demirkesen,A. "MATLAB ile Paralel Hesaplamalara Giriş İnternet Semineri",FİGES A.Ş. (2009)

[25] Grama, A., Gupta, A., Karypis, G., Kumar, V., Paralel Computing, Addison Wesley, 2003.