

Atanmış Sistemler ile Bilgisayar Kümesi Tasarımı ve Mevcut Sistemler ile Karşılaştırması

(Design of a Cluster Computer Using Dedicated Systems and Comparison of Them with the Recent Systems)

Mustafa Haluk AKGÜNDÜZ
İTÜ Fen Bilimleri Enstitüsü
akgunduz@itu.edu.tr

Eşref ADALI
İTÜ Bilgisayar ve Bilişim Fak.
adali@itu.edu.tr

Öz

Çalışmada, paralelleştirmeye uygun olan bir uygulamanın bilgisayar kümeleri mimarisi referans alınarak geliştirilen merkez yönetim yazılımı vasıtasıyla atanmış bilgisayarlarda çalıştırılması gerçekleştirilmiştir.

Güvenilirliğin artırılmasına yönelik olarak havuz yedekleme yöntemi kullanılmıştır. Diğer taraftan verimi artırmaya yönelik olarak ise, (1) dosya aktarımlarında, MD5 bazlı kontroller konularak dosya aktarımı optimize edilmiş ve aynı dosyaların tekrarlı aktarımı engellenmiş, (2) tasarlanan önceliklendirme algoritması ile düğümler önceliklendirilmiş, böylece düğümlerde çalıştırılacak uygulama alt görevlerine öncelik değeri atama imkanı verilmiş, (3) düğümlerin homojen kullanımı sağlanarak yaşlanma oranları dengelenmiş, böylece sistemin ilk bozulma anı ötelenmiştir.

Sistemin sınanması iki farklı bilgisayar ve dört farklı atanmış bilgisayar kullanılarak gerçekleştirilmiştir. Yapılan maliyet analizinde, referans bilgisayar ile atanmış bilgisayarların başarımları / fiyat oranları karşılaştırılmış, atanmış bilgisayarların referans bilgisayara alternatif olup olmadıkları sınanmıştır. Sınama sonuçlarında, referans bilgisayara göre sırasıyla 58%, 289%, 156% ve 314%'lük başarımları / fiyat oranları elde edilmiştir. Buna göre ilk versiyon Raspberry PI atanmış bilgisayarı hariç, tüm atanmış bilgisayarlar

referans bilgisayardan daha iyi başarımları / fiyat oranı göstermişlerdir.

Anahtar Sözcükler: atanmış bilgisayarlar, dağıtık sistemler, bilgisayar kümeleri, beowulf

Abstract

In the study, it is aimed to run a large-scale application, which is suitable for parallelization at the embedded computers with using central management software that taking computer cluster architecture as reference.

Pool backup approach is used to increase system reliability. On the other hand, system efficiency was increased by (1) optimizing file transfers by avoiding redundant transfers via MD5 based controls, (2) using the scoring algorithm designed as part of this study to prioritize the nodes providing a mechanism to assign priority scores to subtasks to be executed at nodes, (3) homogenizing use among all nodes to equalize weathering rates and delay the first breakdown.

The proposed system was tested with two computers and four different types of dedicated computers. A cost analysis was conducted where the cost / performance ratios of the embedded computers were compared to the reference computer to see if these embedded computers can be considered as an alternative to the reference computer. The resulting cost / performance ratios were found to be 58%, 289%, 156% and 314%, respectively. Based on these results, all embedded computers except for first version of Raspberry PI have better cost / performance ratios compared to the reference computer.

Key words: dedicated computers, distributed systems, computer clusters, beowulf

Gönderim ve kabul tarihi : 12.05.2015-09.06.2015

1. Giriş

Bilgisayar dünyasının en önemli sorunlarından biri işlem süresini olabildiğince kısaltmaktır. Bu amaçla merkezi işlem biriminin hızını arttırmak, belleğe erişim süresini kısaltmak ilk hedeftir. Bu hedefe erişmek yarı iletken teknolojisindeki gelişmelerle sağlanmaya çalışılmaktadır ancak sınırlara yaklaşıldığı da bilinmektedir. Bilgisayar hızını arttırmaya yönelik çalışmalardan biri bilgisayar içindeki merkezi işlem birimi sayısını arttırmaktır. Bu yönetime çok çekirdekli mimari adı verilmektedir. İkinci bir çözüm aynı görevin değişik veriler üzerinde yapıldığı uygulamalar için gevşek paralel yapıların kullanılmasıdır. Bu tip uygulamalarda belli veri kümeleri için ayrı ayrı bilgisayarlar atanmaktadır. Çalışmamızda maliyeti düşük, başarımlı kabul edilebilir düzeyde olan bilgisayarlarla gevşek paralel mimari için çözüm araştırılmış ve tasarım gerçekleştirilmiş, deney sonuçları açıklanmıştır.

Özellikle büyük ölçekli uygulamaların paralelleştirilmesi ve bu uygulamaların sunucu görevi gören çok çekirdekli bilgisayarlarda çalıştırılması, geliştirilen uygulamaların işlem bitirme zamanını önemli ölçüde azaltmaktadır. Ancak hem üzerinde çalışılan verinin boyutunun önemli oranda artması hem de uygulamaların sahip olduğu bileşenlerin dışarıyla olan etkileşimlerinin sistemi yavaşlatma olasılığından dolayı hedeflenen işlem zamanı yakalanamamaktadır.

Bunun yanında büyük hacimli işlemlerin belirli bir sürede bitirilmesine yönelik olarak tasarlanan sunucu bilgisayarları piyasadaki sayıları da göz önüne alındığında herkesin ulaşabileceği fiyatların çok ötesinde konumlandırılmış, buna karşılık beklenen başarımın yine de gerisinde kaldıkları gözlenmiştir. Ayrıca sunucu bilgisayarların uzun süreli çalışmalarından kaynaklanan ısı vb. kaynaklı olası sorunlar yapılan işin tamamının belirli sürede askıda kalmasına neden olmakta, çözüme yönelik olarak yapılan periyodik bakımlar ek maliyet oluşturmaktadır.

Sonuç olarak büyük ölçekli işlemlerin belirli bir zamanda bitirilebilmesi mevcut çok çekirdekli sunucu tabanlı çözümler yerine gevşek paralel mimarileri vasıtasıyla birden fazla bilgisayara ağ üzerinden yayılarak gerçekleştirilmeye çalışılmaktadır. Gevşek paralel mimari, kendi içinde birçok alt alana ayrılmış olup, uygulamaların

çalıştırılma süreçlerini farklı noktalardan ele alıp gerekli iyileştirmeleri sağlamaktadır. Büyük ölçekli uygulamaların, gevşek paralel mimarilerin alt alanlarından biri olan bilgisayar kümeleri mimarisi ile tasarlanmış bir atanmış sistemler bütünü vasıtasıyla belirgin bir başarımlı / fiyat üstünlüğüyle çözülebileceği fikri oluşmuştur.

Çalışmada, kendini farklı veri kümeleriyle çok sayıda tekrar eden alt görevlere sahip büyük ölçekli örnek bir uygulamanın, doğası gereği paralelleştirmeye uygun olmasının üstünlüğünü de kullanarak, bilgisayar kümeleri mimarisi vasıtasıyla ayrıştırılıp atanmış bilgisayarlarda çalıştırılmasını sağlamaktadır.

Atanmış bilgisayarlar, belirli bir görevi gerçekleştirmek üzere, üzerinde sınırlı sayıda bileşene ve bireysel bilgisayarlara oranla düşük işlemci gücüne sahip, boyutça küçük bilgisayarlardır. Sahip oldukları bileşenlerin sınırlı sayıda olması ve düşük hızda çalışmaları, harcadıkları gücün az olmasını sağlamakta, bu sayede fansız tasarımlara sahip olabilmektedirler. Atanmış bilgisayarların işlemci gücü her ne kadar bireysel bir bilgisayarın altında kalsa da sağladığı başarımlı ile çok ciddi bir fiyat üstünlüğü sağlamaktadır. Bu nedenlerden dolayı yeterli sayıda atanmış bilgisayarın birlikte çalıştırılmaları ile sorunun çözümünde hem hız hem de ekonomik üstünlük sağlayacağı sonucuna varılmıştır.

Tasarlanan mimaride atanmış bilgisayarların kullanılması, hem görev ve veri tabanlı paralelleştirilmiş uygulamaların hem de atanmış bilgisayarların sahip olduğu işlemci mimarisine göre de dizi ya da vektör paralellliğini kullanan uygulamaların çalışmasına imkan vermektedir. Ayrıca atanmış bilgisayarların birlikte ve bağımsız çalıştırılması sistemin bütününe ilişkin sürdürülebilirlik üstünlüğü de sağlamaktadır. Şöyle ki, atanmış bilgisayarların herhangi bir ya da birkaç tanesinde meydana gelebilecek fiziksel bir sorun otomatik olarak diğerleri tarafından yedeklenebilmektedir, böylece uygulama her türlü dış etkene rağmen başarılı bir şekilde çalışabilmektedir. [1]

Çalışmada bilgisayar kümeleri mimarisi [4] referans alınmış, bilgisayarlar arası haberleşme arabirimi olarak ise soket mimarisi bazlı mesaj aktarım mimarisi kullanılarak dağıtık bir yazılım uygulaması (yönetim yazılımı) tasarlanmıştır. Sonrasında örnek bir uygulamanın bu mimari vasıtası ile ayrıştırılıp atanmış bilgisayarlarda çalışması sağlanıp

başarımları ölçülmüş, elde edilen sonuçlar yüksek işlemci gücüne sahip bilgisayarlarda elde edilen sonuçlarla karşılaştırılmıştır.

1.1. Beowulf Kümesi

Beowulf, [5] paralel hesaplamalarda kullanılmak üzere bilgisayar kümeleri mimarisini referans alan çok bilgisayarlı bir mimari örneğidir. Genellikle bir sunucu bilgisayarı ile çok sayıda istemci bilgisayarının birbirine yerel bir ağda Ethernet ya da başka bir ağ arabirimi vasıtasıyla bağlandığı bir sistemden oluşur. Mimarinin ortaya çıkış amacı, sistemin kolay ulaşılabilen ve maliyeti düşük donanım / yazılım bileşenleriyle oluşturulmasına imkan vermesidir. Kümeyi oluşturan bilgisayarlar için herhangi bir özel donanım bileşeni gerekmemekte, standart Linux, FreeBSD gibi ücretsiz işletim sistemleri kullanılabilir. Beowulf kümesini diğer küme mimarilerinden ayıran bir özellik de istemci rolündeki bilgisayarların oldukça yalın olmaları ve sisteme olan katkılarının dışarıdan sadece ek işlemci ve bellek desteği olarak gözükmesidir.

Çalışma kapsamında seçilen donanımların düşük maliyetli oluşu, birbirleri arasındaki bağlantıların Ethernet arabirimi vasıtasıyla sağlanması ve Unix tabanlı işletim sistemlerini çalıştırabilmesi, tasarımın; Beowulf Kümesi mimarisine yakın olduğunu göstermektedir ancak çalışmamızda özgün bir mimari tasarım geliştirilmiştir.

Çalışma kapsamında seçilen donanımların düşük maliyetli oluşu, birbirleri arasındaki bağlantıların Ethernet arabirimi vasıtasıyla sağlanması ve Unix tabanlı işletim sistemlerini çalıştırabilmesi, tasarımın; Beowulf Kümesi mimarisine yakın olduğunu göstermektedir ancak çalışmamızda özgün bir mimari tasarım geliştirilmiştir.

2. Literatür Araştırması

Bilgisayar kümeleri mimarisi, geçmişte çok uzun zaman süreceği düşünülerek gerçekleştirilemeyeceği düşünülen uygulamaların gerçekleştirilmesini mümkün kılan bir yapı sunmasıyla beraber, çok çeşitli alanlarda kendisine kullanım sahası bulmuştur. Uygulama sahaları ne kadar farklı olursa olsun, paralelleşmeye yatkın uygulamalar birbirlerinden seri çalışması gereken kısımların oranı ve türü noktasında ayrılmaktadırlar. Örnek olarak incelenen araştırmalardan bazıları şunlardır;

Biyomedikal mühendisliğinde paralel ve dağıtık işleme teknikleri: Biyomedikal alanında paralel dağıtık işleme tabanlı yapılan bu çalışmada [16], işlenecek olan resimler "sahip - köle" olgusunda sahip olarak tanımlanan bilgisayar tarafından bağımsız olarak kölelere dağıtılmıştır.

Cluster Juggler - Bilgisayar kümesi tabanlı sanal gerçeklik: Sanal Gerçeklik alanında yapılan bu çalışmada [18], paylaşımlı bellek modelinde çalışan bir sanal gerçeklik uygulaması olan "VR Juggler" uygulaması, bilgisayar kümeleri mimarisi vasıtasıyla hızlandırılmaya çalışılmıştır.

Zemin-Yapı etkileşimi sorunlarının bilgisayar kümeleriyle paralel çözümü: İnşaat Mühendisliği alanında yapılan bu çalışmada [19], Zemin-Yapı Etkileşimi sorunlarına dinamik çözüm üreten örtük Newmark integral algoritmasının paralelleştirilmesi ve bilgisayar kümeleri vasıtasıyla hızlı çözüm üretilmesi sağlanmıştır.

RPiCluster - Raspberry Pi tabanlı Beowulf kümesi: Matematiksel işlemlerin hızlandırılması amaçlı yapılan bu çalışmada [20], Raspberry PI donanımları kullanarak Beowulf Kümesi gerçekleştirilmiş ve başarımları değerlendirilmiştir. Başarım sonuçlarında oluşturulan kümenin mevcut sisteme iyi bir alternatif olmadığı ifade edilmiştir.

Beowulf kümesi üzerinde paralel SAR işlemcisinin geliştirilmesi: Sentetik Aralıklı Radar (SAR) teknolojisi üzerine yapılan çalışmada [24], SAR için geliştirilen G2 ardişlemcisine paralel çalışabilme özelliği kazandırılmaya çalışılmıştır.

3. Merkez Yönetim Yazılımı

Merkez Yönetim Yazılımı, bilgisayar kümeleri mimarisinde, kümelere iş aktarım kontrolünün tek merkezden olmasını sağlayan, kümeler üzerindeki haberleşme, dengeleme, yedekleme gibi işleri yöneten ve düğümlerin yalın bir şekilde kendisine atanan görevi yerine getirmesini sağlayan yazılımdır. Çalışma kapsamında bilgisayar kümeleri mimarisinin gereksinimlerine uygun bir merkez yönetim yazılımı geliştirilmiştir.

Sınama uygulamasının paralelleştirilmiş alt görevleri, belirli bir kural dahilinde merkez yönetim yazılımına tanıtılması gerekmektedir. Yazılım, tanımlanacak bir kural vasıtasıyla uygulamaya ilişkin alt görevleri düğümlere aktaracaktır. Merkez yönetim yazılımının, farklı bilgisayarlarda bulunabilecek, paralelleştirilmiş birden fazla uygulamayı aynı anda işleyip, kontrolündeki düğümlere aktarması gerekmektedir. Bunun için yazılım içerisinde ayrı ve bağımsız çalıştırılabilen alt bileşenlere sahip olacak ve bu bileşenler

vasıtasıyla uygulamalar işletilmek üzere düğümlere aktarılacaktır.

Sistem genelindeki saydamlığı sağlamak için, yazılım; paralelleştirilen uygulamaya ilişkin içerik ya da çalıştırılma işlemleri ile ilişkili herhangi bir süreçte yer almayacak, bu konudaki tüm sorumluluk, uygulama geliştiricisinin sisteme sağladığı kurallara ait olacaktır. Başarıma ilişkin sınama ve karşılaştırmalar için yazılım ölçümle ilişkili alt görevler ile düğümlerin başarımları ve sürdürülebilirliklerine ilişkin raporlama kabiliyetine sahip olacaktır. Yazılım, kontrolünü sağladığı düğümlerin yedeklenmesi ve görevlerin dengeli dağıtılmasından sorumlu olacak, düğümlerin kullanım istatistiklerini tutacaktır.

3.1 İş Akışı

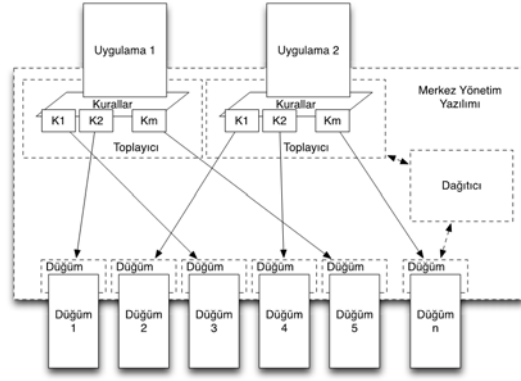
Sürecin belli bir başarımları sağlaması için geliştirilecek yazılım; tek noktadan tam merkezi yönetim yerine, bağımsız çalışabilen alt bileşenleri vasıtasıyla, bileşenlerini farklı düğüm ya da bilgisayarlarda konumlandırarak, yoğun ağ haberleşmesi ve senkronizasyon gibi olumsuz etkilerden uzak işleyişini gerçekleştirecektir. Merkez Yönetim Yazılımı, uygulamanın ayrıştırılarak düğümlerde çalıştırılmasına ilişkin tüm süreci dağıtıcı, toplayıcı(lar) ve düğüm(ler) den oluşan üç ana bileşenle gerçekleştirmektedir.

Dağıtıcı tüm süreci yönetmek yerine sadece toplayıcı ve düğümleri yönetmekte, böylece hem tüm sürecin yönetimi kolaylaşmakta hem de yazılım içerisinde bileşen bazlı yalıtım sağlanmaktadır. Toplayıcı ve düğümler arası aktarım dağıtıcıdan bağımsızlaştığı için haberleşme ve senkronizasyon gibi ek işler ortadan kalkmıştır.

Toplayıcı, uygulamanın bulunduğu bilgisayarda çalışıp, uygulamaya ilişkin paralelleşmiş alt görevlerin tanımlandığı kural dosyalarını çözmekte ve alt görevlerin düğümlere aktarımını sağlamaktadır.

Düğümler ise toplayıcılardan gelen kural ve kuralla ilgili dosyaların çalıştırılmasını sağlamaktadırlar.

Bileşenlerin ilk çalıştırılmaları, bilgisayarların ve düğümlerin açılmasıyla otomatik tetiklenmekte ve ilklendirilmeleri bağımsız olarak gerçekleşmektedir. Merkez yönetim yazılımının işleyişi Şekil-1'de gösterilmiştir.



Şekil-1: Merkez Yönetim Yazılımının İşleyişi

Uygulama alt görevlerinin düğümlere aktarılıp çalıştırılmasına ilişkin süreç aşağıdaki aşamalarla sağlanmaktadır;

1. Dağıtıcı bulunduğu haberleşme ortamına "**Uyandır**" mesajı gönderir ve bunu belirli aralıklarla yapmaya devam eder. Böylece sisteme yeni toplayıcı ve düğümlerin dinamik olarak giriş ve çıkışı sağlanmış olur.
2. Uyandır mesajını alan toplayıcı ve düğümler dağıtıcıya geri "**Hazır**" mesajı gönderirler ve mesaj bekleme kipinde beklemeye başlarlar.
3. Düğümlerden gelen hazır mesajını alan dağıtıcı, düğümleri "boşta" olarak işaretleyip düğüm listesine ekler. Daha önce hazır mesajı aldığı ama son uyandır mesajında cevap alamadığı düğümleri ise listeden siler. Belirlenen güvenilirlik değerine göre yedekte tutulacak düğüm sayısını hesaplar. Bu noktadan sonra mesaj bekleme kipinde toplayıcıdan düğüm istek mesajı bekler.
4. Kullanıcılar tarafından toplayıcıların sağladığı kullanıcı arayüzü vasıtasıyla uygulamanın çalıştırılması süreci başlatılır. Bu aşamada kullanıcı uygulamaya ilişkin kurallar ve kurallarla ilişkili dosyaların, dosya sistemindeki yolunu toplayıcıya tanıtır ve kuralların işletilmesi süreci tetiklenir.
5. Toplayıcı önce dağıtıcıya "**Zaman**" mesajı göndererek, dağıtıcının süreçten haberdar olması sağlar. Dağıtıcı zaman mesajını

uygulama iki yönlü olarak çalışacaktır;

- İlk çalışma kipinde uygulamanın komut satırından değiştirge almadığı ya da sadece tek bir dosyayı değiştirge aldığı durumdur. Bu çalışma kipi uygulamanın yukarıda bahsi geçen, bir dizindeki tüm ya da değiştirge girilen matris dosyalarının içeriklerinin çarpılarak sonucun kabuğa basıldığı durumdur.
- İkinci çalışma kipinde ise uygulama, rastgele ya da değiştirge girilen büyüklüklerde içerisinde iki matrisin yer aldığı dosya üretmektedir. Matrislerin içerdiği değerler ise 1 ile 600 arasındaki sayılardan oluşmaktadır.

4 Sınamalar ve Sonuçlar

Geliştirilen merkez yönetim yazılımı, daha önce değinilen donanımlar kullanılarak sınanmış ve çalışmanın, benzer çalışmalara oranla sağladığı katkı noktasında üstün ve eksik yönleri irdelenmiştir.

4.1 Platform

Seçilen platformlarda işletim sistemi olarak, sunucu işlevi gören bilgisayarlarda Mac OS Yosemite, düğümlerde Linux türevleri tercih edilmiştir. Her iki işletim sisteminin de Unix tabanlı olması, geliştirilen uygulamanın hem x86 hem de ARM mimarilerinde derlenmesi işlemlerinde büyük kolaylık sağlamıştır.

Dağıtıcı, toplayıcı ve düğüm olarak sınanmak üzere,

- Masaüstü bilgisayar:
İşlemci: 4x i7 3.4 Ghz
Bellek: 16 GB DDR3

Sadece düğüm olarak sınanmak üzere,

- Raspberry PI Model B [11]: (3 Adet)
İşlemci: ARM11 700 Mhz
Bellek: 512 MB SDRAM
- Raspberry PI 2 Model B [11]: (1 Adet)
İşlemci: 4x ARM Cortex A7 900 Mhz
Bellek: 1 GB DDR2
- BeagleBone Black Model C [12]: (1 Adet)
İşlemci: ARM Cortex A8 1 Ghz

Bellek: 512 MB DDR3

- HardKernel ODROID U3 [13]: (1 Adet)

İşlemci: 4x ARM Cortex A9 1.7 Ghz

Bellek: 2 GB DDR2

donanımları tercih edilmiştir.

4.2 Uygulama

Yapılan çalışmada, sınama uygulaması olarak 1 ile 600 arasında rasgele değerler alan, rastgele uzunlukta matrislerin yer aldığı dosyalar, düğümlerde çalıştırılıp, düğümlerin başarımları ölçülmüştür. Matris uygulaması vasıtasıyla 20 adet matris girdisine sahip dosya oluşturulmuştur. Sınama uygulamasından beklenen, bütün dosyalardaki matris girdilerin hesaplamasının yapılıp bir çıkış dosyasına yazılmasıdır. Düğüme aktarılma işleminde işlemin tekrar sayısı, kullanıcı arayüzü üzerinden ayarlanabilmektedir. İlk aktarım matris çarpım işlemini gerçekleyen uygulama ile, matris girdilerinin yer aldığı dosyalar iken sonraki aktarımlarda geliştirilen yazılımın sahip olduğu MD5 kontrol mekanizması sayesinde uygulama dosyasının tekrar gereksiz aktarımı engellenmektedir.

Uygulamanın aktarım sürecinden sonra, düğümde çalışan uygulama sırayla tüm matris girdi dosyalarını okuyup çözümlenmekte ve çarpım işlemini gerçekleştirmektedir. Oluşan sonucu ise bir çıkış dosyasına yazmaktadır.

4.3 Sınamalar

Çalışmada ana bileşenlerin sınaması, hem aynı bilgisayar hem de farklı bilgisayarlar kullanılacak şekilde kurgulanmış ve sınama sonuçları elde edilmiştir. Sınama ortamları iki ana yapılandırmadan oluşturulmuştur:

İlk uygulama: Dağıtıcı, toplayıcı ve düğüm rolleri aynı masaüstü bilgisayar üzerinde gerçekleştirilmiştir.

İkinci uygulama: Dağıtıcı ve toplayıcı rolleri masaüstü bilgisayar, düğüm rolleri ise atanmış bilgisayarlar ile gerçekleştirilmiştir.

Ana bileşenlerin aynı bilgisayarda olduğu durumlarda, birbirleri arasındaki haberleşme yöntemi olarak Unix Soketleri, farklı bilgisayarlarda olduğu durumlarda ise İnternet Soketleri tercih edilmiştir. Aktarımın sisteme getirdiği etkiyi en aza indirmek

için sına ma tekrarları aynı matris girdileriyle yapılmış, dolayısıyla düğümle rde çalıştırılacak uygulamaya ek olarak matris girdilerinin de ilk aktarım dışında gönderimi engellenmiştir. Tablo gösteriminde kolaylık olması açısından,

Tekrar: Matris çarpım işleminin düğümle r tarafından toplamda kaç kere tekrarlanacağını temsil etmektedir.

İş Süresi: Her bir alt görevin düğümde başlaması ve bitmesi arasında geçen süreyi

Toplam Süre: Toplayıcının süreci başlatmasıyla beraber, aktarımlar ve mesajlaşmalar dahil geçen toplam süreyi temsil etmektedir.

Sınama 1

Yapılandırma:

Dağıtıcı : Masaüstü Bilgisayar

Toplayıcılar : Masaüstü Bilgisayar

Düğümle r : Masaüstü Bilgisayar

Tek düğümlü yapılan sına manın sonucu Çizelge-1.1'de, iki düğümlü yapılan sına manın sonucu Çizelge-1.2'de ve üç düğümlü yapılan sına manın sonucu Çizelge-1.3'de gösterilmiştir.

Çizelge-1.1: Sınama 1 - Tek Düğümlü

Tekrar	Düğüm	İş Süresi	Toplam Süre
1	Masaüstü	1.102 s	1.448 s
2	Masaüstü	0.693 s	2.180 s
3	Masaüstü	0.710 s	2.930 s
4	Masaüstü	0.710 s	3.680 s
5	Masaüstü	0.745 s	4.465 s
6	Masaüstü	0.744 s	5.249 s

Çizelge-1.2: Sınama 1 - İki Düğümlü

Tekrar	Düğüm	İş Süresi	Toplam Süre
1	Masaüstü	1.178 s	1.625 s
2	Masaüstü	1.183 s	1.629 s
3	Masaüstü	1.040 s	2.721 s
4	Masaüstü	1.040 s	2.721 s
5	Masaüstü	1.180 s	3.953 s
6	Masaüstü	1.180 s	3.953 s

Sınama sonucundan da görüleceği üzere tek düğümlü

sisteme göre iki düğümlü sistemin başarı m katkısı yaklaşık %25 iken düğüm sayısı üçe çıktığında bu katkı %18'te kalmıştır.

Çizelge-1.3: Sınama 1 - Üç Düğümlü

Tekrar	Düğüm	İş Süresi	Toplam Süre
1	Masaüstü	1.859 s	2.199 s
2	Masaüstü	1.861 s	2.201 s
3	Masaüstü	1.861 s	2.201 s
4	Masaüstü	1.399 s	4.140 s
5	Masaüstü	1.535 s	4.276 s
6	Masaüstü	1.562 s	4.303 s

Sınama 1 sonucu göstermektedir ki, aynı bilgisayarda düğüm sayısını ne kadar artırırsak artıralım, simüle edilen düğümle rin tek işlemciyi paylaşım lı olarak kullanmaları ve girdi dosyalarını dosya sisteminden ancak sıralı okuyabilmeleri gibi birlikte çalışmayı engelleyen seri çalışma zorunluluklarından dolayı;

$$\frac{\text{Toplam Başarı m}}{\neq} \times \text{Düğüm Sayısı} = \text{Düğüm Başarı m}$$

olmaktadır.

Sınama 2

Yapılandırma:

Dağıtıcı : Masaüstü Bilgisayar

Toplayıcılar : Masaüstü Bilgisayar

Düğümle r : Raspberry PI Model B (1 - 3)

Tek düğümlü yapılan sına manın sonucu Çizelge-2.1'de, iki düğümlü yapılan sına manın sonucu Çizelge-2.2'de ve üç düğümlü yapılan sına manın sonucu Çizelge-2.3'de gösterilmiştir.

Çizelge-2.1: Sınama 2 - Tek Düğümlü

Tekrar	Düğüm	İş Süresi	Toplam Süre
1	Raspberry PI #1	22.303 s	22.497 s
2	Raspberry PI #1	21.949 s	44.699 s
3	Raspberry PI #1	22.089 s	67.039 s
4	Raspberry PI #1	21.816 s	89.132 s
5	Raspberry PI #1	21.996 s	111.585 s
6	Raspberry PI #1	21.553 s	133.577 s

Çizelge-2.2: Sınama 2 – İki Düğümlü

Tekrar	Düğüm	İş Süresi	Toplam Süre
1	Raspberry PI #1	23.594 s	24.055 s
2	Raspberry PI #2	24.098 s	24.561 s
3	Raspberry PI #1	22.780 s	47.092 s
4	Raspberry PI #2	23.077 s	47.898 s
5	Raspberry PI #1	22.292 s	69.755 s
6	Raspberry PI #2	23.393 s	72.069 s

Bu sınamada ortaya çıkan sonuç, Raspberry PI donanımı her ne kadar masaüstü bilgisayarının %4'ü kadar başarımlı gösterse de, düğümlerin birbirinden bağımsız oluşu ve seri çalışmaya yönelik en önemli kısıtın, uygulamanın ağ üzerinden düğüme aktarım işlemi gibi göz ardı edilebilecek bir işlem olması; düğümlerden elde edilen başarımların toplamdaki başarımla etkisinin, düğümlerin sayısı ile doğru orantılı göstermesini sağlamıştır.

Sınama sonucuna göre, ikinci düğümün eklenmesi sistemde %92, üçüncü düğümün eklenmesi ise %199'luk bir başarımların artışı sağlamıştır. Sonuç olarak düğümlerin gösterdiği toplam başarımların, beklenene yakın olarak gerçekleşmektedir;

$$\text{Toplam Başarımlar} \approx \text{Düğüm Sayısı} \times \text{Düğüm Başarımları}$$

Çizelge-2.3: Sınama 2 - Üç Düğümlü

Tekrar	Düğüm	İş Süresi	Toplam Süre
1	Raspberry PI #3	21.860 s	22.501 s
2	Raspberry PI #1	22.053 s	22.702 s
3	Raspberry PI #2	22.266 s	22.880 s
4	Raspberry PI #3	21.280 s	44.021 s
5	Raspberry PI #1	21.470 s	44.414 s
6	Raspberry PI #2	21.500 s	44.611 s

Sınama 3

Yapılandırma:

Dağıtıcı : Masaüstü Bilgisayar

Toplayıcılar : Masaüstü Bilgisayar

Düğümler : Raspberry PI 2 Model B

Tek düğümlü yapılan sınamanın sonucu Çizelge-3'te gösterilmiştir.

Çizelge-3: Sınama 3 - Tek Düğümlü

Tekrar	Düğüm	İş Süresi	Toplam Süre
1	Raspberry PI v2	3.864 s	3.891 s
2	Raspberry PI v2	3.541 s	7.540 s
3	Raspberry PI v2	3.591 s	11.234 s
4	Raspberry PI v2	3.519 s	14.857 s
5	Raspberry PI v2	3.579 s	18.548 s
6	Raspberry PI v2	3.837 s	22.490 s

Raspberry PI 2'nin başarımlarını, sınanan ilk versiyonuna göre yaklaşık %565 daha iyi olsa da, masaüstü bilgisayarın başarımlarının yaklaşık %21'i kadar bir başarımlara karşılık gelmektedir.

Sınama 4

Yapılandırma:

Dağıtıcı : Masaüstü Bilgisayar

Toplayıcılar : Masaüstü Bilgisayar

Düğümler : BeagleBone Black Model C

Tek düğümlü yapılan sınamanın sonucu Çizelge-4'te gösterilmiştir.

Çizelge-4: Sınama 4 - Tek Düğümlü

Tekrar	Düğüm	İş Süresi	Toplam Süre
1	BeagleBone Black	7.461 s	8.023 s
2	BeagleBone Black	7.179 s	15.343 s
3	BeagleBone Black	7.131 s	22.615 s
4	BeagleBone Black	7.240 s	29.997 s
5	BeagleBone Black	7.203 s	37.343 s
6	BeagleBone Black	7.237 s	44.721 s

BeagleBone Black, giriş / çıkış özellikleri anlamında sınanan donanımlar arasında en zengin özellik yelpazesine sahip olsa da, işlemcisinin tek çekirdekli

olması, beklenen başarıyı verememesine neden olmuştur. İlk versiyon Raspberry PI'ye oranla %300 daha iyi başarıyı sergilemiş olsa da, ikinci versiyon Raspberry PI'nin %53'ü ve masaüstü bilgisayarın %11'i kadar başarıyı sergileyebilmiştir.

Sınama 5

Yapılandırma:

Dağıtıcı : Masaüstü Bilgisayar

Toplayıcılar : Masaüstü Bilgisayar

Düğüm : HardKernel ODroid U3

Tek düğümlü yapılan sınamanın sonucu Çizelge-5'te gösterilmiştir.

HardKernel ODroid U3, sınanan donanımlar arasında en başarılı donanım olarak ön plana çıkmıştır. En yakın rakibi Raspberry PI 2'den yaklaşık %224 daha iyi başarıyı sergilemiş, buna karşın masaüstü bilgisayarın %48'i kadar bir başarımda kalmıştır.

Çizelge-5: Sınama 5 - Tek Düğümlü

Tekrar	Düğüm	İş Süresi	Toplam Süre
1	ODroid U3	1.878 s	1.904 s
2	ODroid U3	1.631 s	3.590 s
3	ODroid U3	1.574 s	5.219 s
4	ODroid U3	1.711 s	6.996 s
5	ODroid U3	1.682 s	8.731 s
6	ODroid U3	1.634 s	10.417 s

4.4 Maliyet Analizi

Yapılan sınamalar sonucunda ortaya çıkan başarı değerleri ile sınamaya tabi tutulan bilgisayarlar ve atanmış bilgisayar donanımlarının başarı / fiyat dengeleri karşılaştırılmıştır. Sınamalarda referans alınan donanım masaüstü bilgisayar olmuş, diğer bütün donanımlar masaüstü bilgisayarına oranla karşılaştırmadaki yerlerini almışlardır.

Masaüstü bilgisayar, günümüz piyasasında yapılan fiyat araştırmasında; işlemci ve belleğin sınamaya giren değerlerde kalması koşuluyla, diğer zorunlu donanım bileşenleri en az fiyatlı bileşenlerden seçilmiştir. Buna göre seçilen donanıma ilişkin

bileşen listesi Çizelge-6'da gösterilmiştir.

Sınanan donanımlara ilişkin bileşen ve fiyat bilgisi ise Çizelge-7'de gösterilmiştir.

Çizelge-6: Masaüstü Bilgisayar Bileşenleri

	Bileşen	Fiyat
İşlemci	i7 4x 3.40 Ghz	1051 TL
Bellek	2x8 GB DDR3	547 TL
Anakart	GA-B75M-D2V	226 TL
Sabit Disk	1TB 7200RPM	189 TL
Kasa	400W ATX	155 TL
Toplam		2168 TL

Çizelge-7: Donanım Bileşenleri

Bileşen	Fiyat
Raspberry PI Model B	125 TL
Raspberry PI 2 Model B	125 TL
BeagleBone Black Model C	139 TL
HardKernel ODroid U3	230 TL

Sınanan donanımlara ilişkin başarı ve fiyat dengesi uyarınca çıkarılan maliyet indeksi aşağıdaki gibi ifade edilmiştir;

$$DA = \left[\frac{MP}{DP} + 0.5 \right] \quad (1)$$

DA: Gerekli en az donanım adedi, MP: Masaüstü bilgisayar başarıyı, DP: Donanım başarıyı

$$TDA = DA + [DA \times GY] \quad (2)$$

GY: Havuzda yedekte tutulacak donanımın yüzdesi (Güvenilirlik yüzdesi), TDA: Toplam gerekli donanım adedi

$$Mİ = \frac{MF}{DF \times TDA} \geq 1 \quad (3)$$

Mİ: Maliyet indeksi, MF: Masaüstü bilgisayar fiyatı, DF: Donanım fiyatı

Maliyet İndeksi ne kadar yüksek olursa, seçilen donanımın başarımları / fiyat oranı o kadar iyi olmakta, 1'in üzerinde olması ise, seçilen donanımla oluşturulacak sistemin masaüstü bilgisayardan oluşan sisteme göre daha iyi başarımları / fiyat değerine sahip olacağını göstermektedir.

Belirtilen denklemlere göre oluşturulan maliyet indeksi değerleri %10 güvenilirlik için Çizelge-8'de gösterilmiştir;

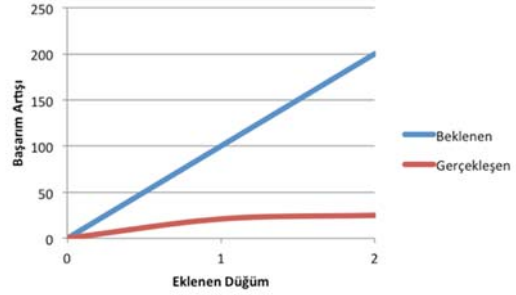
Çizelge-8: Maliyet İndeksi Tablosu

Donanım	MP / DP	TDA	Mİ
Raspberry PI	26.514	30	0.578
Raspberry PI 2	4.689	6	2.891
BeagleBone Black	8.804	10	1.560
ODroid U3	2.086	3	3.142

5. Sonuç

Uygulamanın tek bir bilgisayarda çalıştırılması durumunda, bilgisayarın işlemci başarımları ne kadar iyi olursa olsun sınırlı olduğundan, özellikle çok sayıda paralel ve birbirinden bağımsız alt görevlere sahip uygulamalar işlemciyi paylaşımlı olarak kullanmak zorundadırlar. O nedenle bu tarz uygulamalar daha hızlı çalıştırılma imkanına sahip olsa da işlemci bariyerine takılmaktadırlar. Sistemi genel anlamda engelleyen bir diğer değişirge de uygulamaların kullandığı depolama alanı ve ağ arabirimlerinin tekil oluşudur. Yani uygulamalar depo alanına ve ağ arabirimlerine sıralı erişebilmektedirler, bu da uygulamaların paralel çalışma imkanlarını kısıtlamaktadır.

Masaüstü bilgisayarın tek başına bütün rolleri üstlendiği sınıma sonucunda, tek düğümlü sisteme ikinci bir düğüm eklendiğinde iki katı bir başarımları beklenirken bu başarımları %25'de kalmıştır. Yine benzer şekilde sisteme üçüncü düğüm eklendiğinde başarımları artışı %200 yerine %18'de kalarak iki düğümlü de görülen artışın da altında kalmıştır. Başarımları artışına ilişkin grafik Şekil-6'te gösterilmiştir.

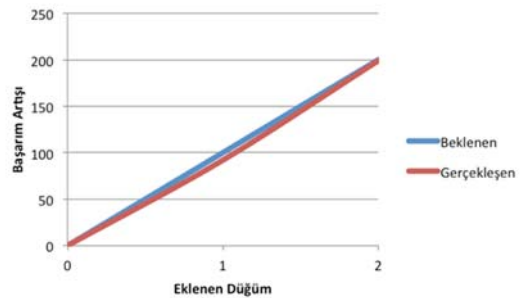


Şekil-6: Tek Bilgisayar - Başarımları Artışı.

Uygulamaların paralel alt görevlerinin birbirlerinden bağımsız olarak atanmış bilgisayar donanımlarında çalışabiliyor olması, her alt görevin herhangi bir bekleme maruz kalmadan olabilecek en yüksek başarımları tamamlanmasına imkan tanımaktadır.

Dolayısıyla, donanımların işlemci gücü ve diğer bileşenlerinden en uygun oranda yararlanılabilmektedir. Bu durumda donanım bileşenlerinin özellikleri uygulamaların çalışma hızına doğrudan etki etmektedir.

Üç adet Raspberry PI donanımlarıyla yapılan sınıma göstermiştir ki, sistemden elde edilen toplam başarımları, donanımların teker teker elde ettikleri başarımları sonuçlarının toplamına yakındır. Tek düğümlü sisteme ikinci bir düğüm eklendiğinde başarımları artışı %92 olmakta, üçüncü düğümün eklenmesiyle beraber artış %199'u bulmaktadır. Başarımları artışına ilişkin grafik Şekil-7'te gösterilmiştir.



Şekil-7: Üç Raspberry PI - Başarımları Artışı.

5.1 Atanmış Bilgisayar Başarım Sonuçları

Sınamaların sınırlı sayıda atanmış bilgisayarla yapılmasından dolayı birbirleri arasındaki karşılaştırmalar tek düğümlü olarak yapılmış ve sonuçları irdelenmiştir;

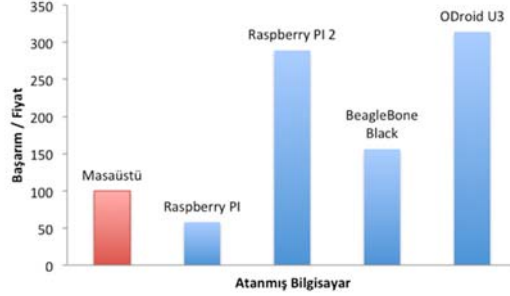
Raspberry PI Model B: İşlemci gücü itibariyle diğer donanımların biraz gerisinde kalan Raspberry PI, tek düğümlü sınamada, sınamayı yaklaşık 134s'de bitirerek, masaüstü bilgisayarın %4'üne karşılık gelen bir başarımlı sergilemiştir. Yapılan maliyet analizine göre, %10 güvenilirlikli çalışma kipinde 30 adet donanım gereksinimiyle masaüstü bilgisayarın %58'ine karşılık gelen bir başarımlı / fiyat oranı sergilemiştir. Gösterdiği başarımlı ile masaüstü sistemine bir alternatif olamamıştır.

Raspberry PI 2 Model B: Önceki modele göre altı kat başarımlı vaat eden Raspberry PI 2, tek düğümlü sınamada, sınamayı yaklaşık 23 s'de bitirerek, masaüstü bilgisayarın %21'ine karşılık gelen bir başarımlı sergilemiştir. Yapılan maliyet analizine göre, %10 güvenilirlikli çalışma kipinde 6 adet donanım gereksinimiyle masaüstü bilgisayarın %289'una karşılık gelen bir başarımlı / fiyat oranı sergilemiştir. Gösterdiği başarımlı ile masaüstü sistemine iyi bir alternatif olmuştur.

BeagleBone Black Model C: Genişleme yetenekleriyle diğer donanımların önüne geçen BeagleBone Black Model C, tek düğümlü sınamada, sınamayı yaklaşık 45 s'de bitirerek, masaüstü bilgisayarın %11'ine karşılık gelen bir başarımlı sergilemiştir. Yapılan maliyet analizine göre, %10 güvenilirlikli çalışma kipinde 10 adet donanım gereksinimiyle masaüstü bilgisayarın %156'sına karşılık gelen bir başarımlı / fiyat oranı sergilemiştir. Gösterdiği başarımlı ile masaüstü sistemine iyi bir alternatif olmuştur.

HardKernel ODroid U3: İşlemci gücü itibariyle diğer donanımların önüne geçen ODroid U3, tek düğümlü sınamada, sınamayı yaklaşık 10 s'de bitirerek, masaüstü bilgisayarın %48'ine karşılık gelen bir başarımlı sergilemiştir. Yapılan maliyet analizine göre, %10 güvenilirlikli çalışma kipinde 3 adet donanım gereksinimiyle masaüstü bilgisayarın %314'üne karşılık gelen bir başarımlı / fiyat oranı sergilemiştir. Gösterdiği başarımlı ile masaüstü sistemine en iyi alternatif olmuştur.

Atanmış bilgisayarların masaüstü bilgisayarına oranla başarımlılarına ilişkin grafik Şekil-8'te gösterilmiştir.



Şekil-8: Atanmış Bilgisayar Başarım / Fiyat Grafiği.

6 Kaynaklar

- [1] Adalı, E. (1980). Dağıtılmış Bilgisayarlarla Denetim *Doçentlik Tezi*, İSTANBUL.
- [2] Rivest, R. (1992). The MD5 Message-Digest Algorithm, *RFC Editor*, United States.
- [3] Ishikawa, Y. ve Tokoro, M. (1984). The Design of an Object Oriented Architecture, *SIGARCH Comput. Archit. News*, 12(3), 178–187.
- [4] Pfister, G., (2000). *Cluster Computing, Encyclopedia of Computer Science*, John Wiley and Sons Ltd., Chichester, UK, s.218–221.
- [5] Meredith, M. ve diğerleri (2003). Exploring Beowulf Clusters, *J. Comput. Sci. Coll.*, 18(4), 268–284.
- [6] Yiqun, C., (2005). Parallel And Distributed Computing Techniques In Biomedical Engineering, *Yüksek Lisans Tezi*, National University Of Singapore.
- [7] Olson, E., (2002). Cluster Juggler – PC cluster virtual reality, *Yüksek Lisans Tezi*, Iowa State University.
- [8] Bahcecioglu, T., (2011). Parallel Solution Of Soil-Structure Interaction Problems On Pc Clusters, *Yüksek Lisans Tezi*, Middle East Technical University.
- [9] Kiepert, J. (2013). RPiCLUSTER - Creating a Raspberry Pi-Based Beowulf Cluster, *Teknik Rapor*, Boise State University.
- [10] Bennett, T., (2003). Development of a Parallel SAR Processor on a Beowulf Cluster, *Yüksek Lisans Tezi*, The University of Cape Town.

- [11] Raspberry PI, <http://www.raspberrypi.org/>, alındığı tarih: 29.03.2015.
- [12] BeagleBone, <http://beagleboard.org/bone>, alındığı tarih: 29.03.2015.
- [13] ODROID, <http://www.hardkernel.com/>, alındığı tarih: 29.03.2015.

