

BİR SANAL PARALEL MAKİNE

B. DURMUŞ* & A. ÖZMEN*

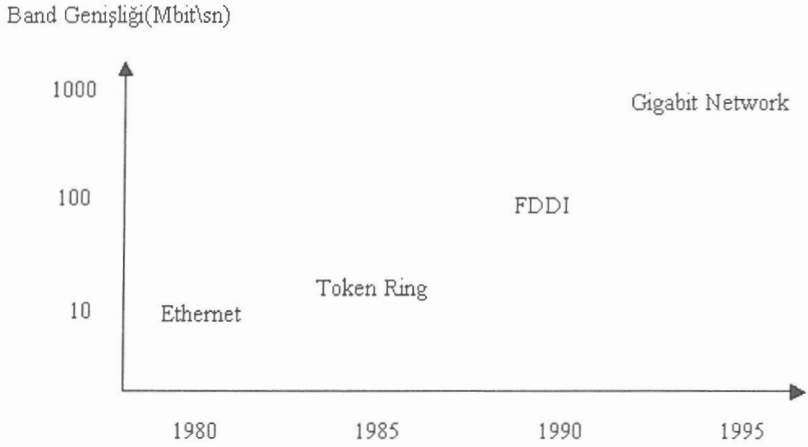
Özet

Bazı uygulamalar için yüksek performanslı bilgisayarlara ihtiyaç vardır. Bu uygulamalar gerçek zamanlı kontrol sistemleri, gerçek zaman kriterlerine esnek ancak yine de zamana karşı hassas olan simulatörler veya icrası interaktif olmadığı halde çok uzun süren hesaplama ağırlıklı programlardır. Bu çalışmada bu tür ihtiyaçlara cevap verebilecek, dağıtık hesaplama mimarisi altında, bir işletim sistemine ek olarak tasarlanmış sanal bir paralel bilgisayar (PVM: Parallel Virtual Machine) ve grafiksel arayüzü XPVM tanıtılmıştır. PVM, birbirlerine ağ ile bağlı, farklı mimari ve işletim sistemleri ile çalışan dağıtık halde duran bilgisayarları bir amaç için ve aynı anda çalıştırmaya yarayan yazılım pakettir. PVM ile sisteme katılan bilgisayarlar görünürde birer iş istasyonu ya da kişisel bilgisayar olsalar da, sistem sonunda sanal bir paralel makineye dönüşmektedir. Dolayısıyla PVM, oldukça pahalı yüksek performanslı bilgisayar sistemleri alamayan üniversiteler ve araştırma kurumları için ekonomik bir çözümdür.

1. Giriş

Bilgisayar donanımı, yazılımı ve algoritmalarındaki hızlı ilerlemelere rağmen büyük ölçekli simülasyonlar, zaman duyarlı hesaplama, bilgisayar destekli tasarım, bilgisayarların imalatta kullanılması ve verinin görüntülenmesi gibi yüksek hesap yeteneği isteyen alanlardan gelen taleplerin artması, mevcut sistemleri ciddi performans kayıplarına sürüklemekte ve yüksek performanslı bilgisayar sistemlerine olan ihtiyacı arttırmaktadır. Bu ihtiyaca cevap verebilmek için ortak bellekli ve dağıtık hesaplama modeli önerilmiştir [2]. Ortak bellekli sistemlerin pahalı olması ekonomik çözümler sunmamaktadır. Buna alternatif dağıtık hesaplama, ekonomiktir ancak programlanması zordur. Sonuç olarak, en iyi maliyet-performans çözümünü ucuz maliyetli işlemcileri tek çatı altında toplamaktır [5].

Saniyede milyonlarca işlem yapan tek işlemcili iş istasyonları yaygındır ve hesap yetenekleri her geçen gün güçlenmektedir. Uygun bir yüksek hızlı ağ ile bu bilgisayarlar bağlandığında müşterek hesap gücü, birçok yoğun uygulamanın çözümünde kullanılabilir. Gerçekte ağ hesaplaması süper bilgisayar düzeyinde hesaplama gücü bile sağlayabilir. Etkin olması için dağıtık hesaplama yüksek iletişim hızları gerektirir. Aşağıda ağ hesaplama hızının yıllara göre gelişimi görülmektedir.



- Ethernet: Popüler yerel alan paket anahtarlama teknolojisi olan Ethernet, 10 Mbit/sn dağıtım özelliğine sahiptir.
- Token Ring: Halka şeklindeki hat üzerinde bir jeton dolaştırılır ve paket bu jeton ile taşınır. 4-16 Mbit/sn hıza sahiptir.
- Fiber Dağıtılmış Veri Arayüzü (FDDI; Fiber Distributed Data Interface): İki istasyon arasında optik fiber iletimi kullanan 100 Mbit/sn token ring modelidir ve güvenilirliği sağlamak için yedek ring içerir.
- Gigabit Ethernet: 1000 Mbps hıza sahiptir. Fiyatı yüksek olduğundan henüz geniş bir alanda kullanılmamaktadır [1].

Tek işlemcili seri bilgisayar için geleneksel programlar yazılmıştır ve birim zamanda bir talimat gerçekleştirilir. İşlemci hızı, verinin donanımda ne kadar hızlı dolaştığına bağlıdır. Elektrik akımının hızı bakır tel limitleri (9 cm/nsn) için düşünülürse; en hızlı bilgisayarlar bir talimatı saniyenin milyarda dokuzu sürede gerçekleştirir [3]. Bir talimatın bu kadar kısa bir sürede gerçekleştirilmesini yeterince hızlı olduğu düşünülebilir ancak, daha hızlı işlem gerektiren birçok problem sınıfı vardır:

- Simülasyon ve modelleme problemleri; başarılı yaklaşımlara dayalıdır, daha fazla hesaplama ve hassasiyet ister.
- Büyük miktarda verinin hesaplamasına dayalı problemler; görüntü işleme, veri tabanı oluşturma, yarı iletken ve süper iletken modellemesi, kuantum mekanikleri, sismik, görünüm v.s. [3].

Yüksek hesap gücü isteyen bu büyük problemlerin çözümü için dağıtık hesaplama modeli cazip hale gelmekte ve bu doğrultuda paralel programlar yazılmaktadır.

Paralel programlama şunları içerir:

- Bir algoritmayı veya veriyi parçalara bölmek
- Farklı işlemcilerle eşzamanlı çalışan görev parçalarını dağıtmak
- İşlemcilerin iletişimini koordine etmek [3].

Paralel programlamada problem parçalara bölünür. Daha sonra bu görev parçaları sisteme bağlı diğer işlemcilerle dağıtılmaktadır. Bu görev, sistemi yöneten bir servis programı tarafından yapılır. Servis programı, dağıtılan görev parçalarının diğer işlemcilerde icrasını başlatır ve görevleri tamamlandığında sonuçları geri toplar. Paralel programlamadaki en büyük zorluk, bütün bu olayları gerçekleyebilecek paralel programların yazılmasıdır.

2. SANAL PARALEL MAKİNE (PVM; PARALLEL VIRTUAL MACHINE)

PVM, bir paralel bilgisayar gibi görünecek şekilde bir ağa bağlanan bilgisayarların heterojen olarak toplanmasına imkan veren bir yazılım paketidir. Hedef, ağa bağlı bilgisayarları bir araya getirip hesap yeteneği güçlü sanal paralel bir bilgisayar oluşturmaktır. Kullanıcının sanal makinasında toplanan donanım tek işlemcili bilgisayarlar, vektör makinaları veya paralel süper bilgisayarlar olabilir. Elemanların hepsi tek tip (homojen) veya farklı tiplerde (heterojen) mimarilere sahip olabilir. PVM sistemi aynı odada bilgisayarları bağlayan bir LAN kadar küçük veya dünya çapında bilgisayarları bağlayan İnternet kadar büyük olabilir. Merkezi bir kontrol altında farklı bilgisayar mimarilerini bir araya getirmeye yönelik bu kabiliyet sayesinde PVM kullanıcısı, bir problemi alt görevlere bölebilir ve her birini icra edilmek üzere en uygun işlemci mimarisine atayabilir. Kullanıcının görevleri prosesleri başlatıp bitirmek, veri paylaşımı ve senkronizasyondur. Görev boyutuyla ilgili bir sınırlama olmamakla birlikte iki bilgisayar arasında veri göndermek için gerekli zaman nedeniyle görevlerin boyutlarına göre PVM'in etkin kullanılması önerilir.

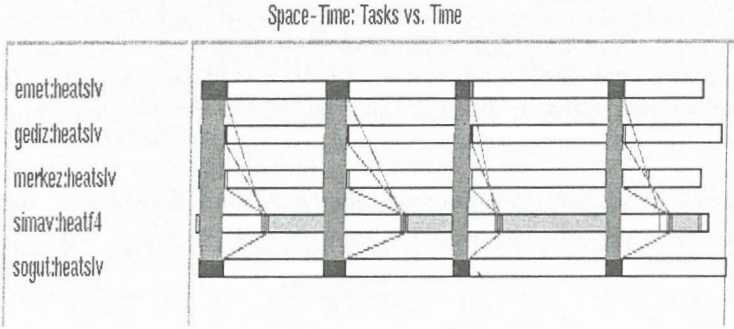
PVM'in ilk sürümü 1989 yazında Oak Ridge Ulusal Laboratuvarı'nda yazıldı. Dağıtım olmadı ama laboratuvardaki uygulamalarda kullanıldı. 2. sürüm Şubat 1991'de Tennessee Üniversitesi'nde yazıldı ve aynı yılın mart ayında dağıtıldı. 1.5 yıl sonra 2. sürüme ilavelere ihtiyaç duyuldu ve 3. sürüm Eylül 1992'de yazılmaya başlandı. Yazılımın ilk kullanımı 1993 Mart'tır. Hali hazırdaki sürüm 3.4 olup müteakip sürümler beklenmektedir. PVM'in yüklenmesi özel hiçbir izin gerektirmez. Ücretsizdir. Yüklenmesi kolaydır ve sadece birkaç Mbyte disk alanı gerektirir. C ve Fortran'ı destekler. Her görev en uygun mimariye atanarak performans optimize edilir [1]. PVM geleceğin hesaplama gereksinimleri için dağıtılmış hesaplama imkanları sunar. Dağıtılmış hesaplama, mevcut donanımı kullanır ve maliyetleri düşük tutar.

Sanal bilgisayar kaynakları aşama aşama büyüyebilir, en son hesap ve ağ teknolojilerinin avantajlarını alabilir. Müşterek çalışmayı kolaylaştırır. PVM, süratle standart hale gelen Mesaj Geçiş Kütüphanesi (MPI:Message Passing Interface) olgusunu kullanmaktadır. Bu olguya göre tüm paralel işlemlerde veri müşterek çalışan görevler arasında değiş tokuş edilmelidir. Standartlaşan bu olgu ile geliştirilen uygulamaların diğer mimarilere taşınması da standart hale gelmektedir. PVM'in bu olguyu kullanması taşınabilirliğini göstermektedir. Paralel program uygulamaları için birçok olanak sunan PVM, dünya çapında kabul görmekte ve büyük problemlerin çözümünde kullanılmaktadır [4].

3. XPVM: PVM İÇİN GRAFİKSEL ARAYÜZ

XPVM, PVM konsol komutlarına grafiksel ara yüz olanağı sunar. PVM programlarının icrası gözlemlenirken ağın durumu, uzay-zaman bakışı ve görev takibi gibi bazı gözlemlenebilir imkanları sunar. Performansın belirlenmesi veya programın hatalardan arındırılması için bu bakış açısı, paralel PVM programında çalışan işlerin birbirleriyle etkileşimini hakkında bilgi kaynağı sağlar.

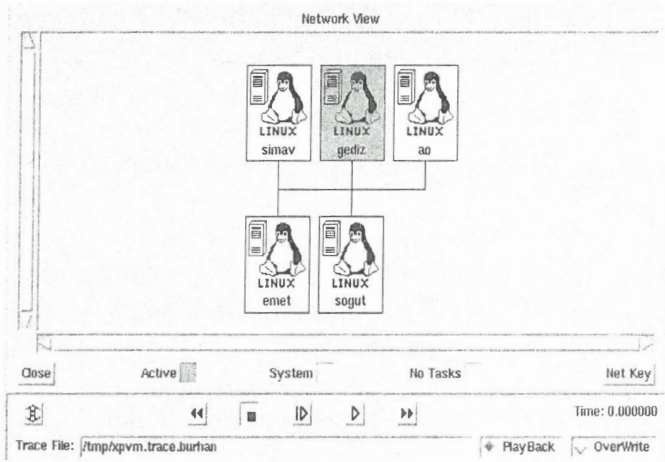
XPVM kullanan bir programda, kullanıcının ihtiyaç duyduğu tek şey 3.3 ya da yukarı versiyon bir PVM kütüphanesi kullanarak programını derlemektir. Zira 3.3 versiyon ve yukarıları çalışma zamanında iz-sürme bilgisi ile donatılmıştır. İz-sürme bilgisi ile kullanıcı gerçek zamanda analiz yapma olanağına sahiptir. Bu özelliğe ait grafik aşağıda görülmektedir.



Şekil 3 Uzay-zaman bakışı

Uzay-Zaman Bakışı, bilgisayarların her bir görevin icrası sırasındaki durumlarını gösterir. Her görev yatay bir çubukla temsil edilir ve çubuğun o anki rengi görevin durumunu belirtir. Makine hesaplama yapıyorsa (computing) görev çubuğun rengi hesaplama rengini alacaktır. Arta kalan (overhead) rengi, işin haber amaçlı yada görev kontrol amaçlı PVM sistem komutları ile meşgul olduğunu temsil eder. Bekleme (waiting) rengi diğer görevlerden mesaj beklediğini gösterir. Mesajlaşma rengi ise mesaj iletimi ve alma işlemlerini temsil eder. İlgili durumlar için kullanılan renkler; hesaplama: yeşil, arta kalan: sarı, bekleme: beyaz ve mesajlaşma: kırmızıdır. Dikey mavi çizgi ise uygulamaya ait en son olayı gösterir.

Sanal makinada terminal (host) üzerinde ağ bakış açısı yüksek seviye aktivite sağlar. Her bir host, ismi ve mimarisi belirlenmiş olacak şekilde bir simge (icon) resmi ile temsil edilir. Bu simgeler her bir hostta çalışan işleri gösterecek şekilde farklı renkler ile gösterilir. "Aktif" renk en azından bir tane işin ilgili hostta yararlı bir görev yaptığını gösterir. "Sistem" rengi kullanıcı ile ilgili bir işlemin yapılmadığını, öte yandan en az bir işin PVM sistem rutinleri ile ilgili icra görevi gördüğünü ifade eder. Terminal üzerinde hiç bir iş yoksa, simge renksiz yada beyaz ile ifade edilir. Terminallerin her bir durum karşısında alacağı renkler belirlidir. Örneğin yeşil aktif terminali, sarı ise sistem işlemlerinin yapıldığını temsil eder. Ancak renk ayarları kullanıcı tarafından yeniden düzenlenebilir.



Şekil 4 Ağ bakış açısı

XPVM, ağa bakış ve uzay-zaman gözlenebilirliğinin dışında işlemci kullanım oranı, programın icrası esnasında mesajlaşma paketlerini ve mesaj yığınlarının durumunu hem on-line hem de off-line olarak gözlemlenmesine imkan sağlar. Performansın belirlenmesi veya programın performans hatalarından arındırılması için bu görsel araçlar vazgeçilmezdir [5]. Sonuç olarak XPVM'in sunduğu gözlenebilirlik imkanları, paralel programın icrası esnasındaki olayların takibi, programların ve hesaplama ortamının kontrolü ve performans analizini daha kolay kılmaktadır.

4.SONUÇ

Paralel hesaplama olanağı giderek büyümektedir. Bu talebe hem yüksek performans hem de ekonomik çözümler sunmak gerekir. PVM, mimari heterojenliği ve en son teknolojilere adaptasyonu ile kaynakların etkin kullanımını gerçekleştirmekte, paralel hesaplama gereksinimlerine ekonomik bir çözüm sunmaktadır. Öte yandan, pahalı hesaplama kaynaklarının verimli kullanılabilmesi için paralel programlar mutlaka performans monitörü ile izlenmelidir. Performans ölçümleriyle paralel sistemler daha verimli çalışacak şekilde yönetilebilirler veya paralel programlar değişik giriş ve işlemci konfigürasyonlarında daha verimli çalışacak şekilde ayarlanabilirler [5]. Sonuç olarak, XPVM'in sunduğu gözlenebilirlik, performans analizine ve paralel hesaplamaların izlenebilirliğine yardımcı olmaktadır.

KAYNAKLAR

[1] Geist A., Begueclin A., Dongarra J., Jiang W., Manchek R., Sunderman V. : PVM: Parallel Virtual Machine- A User's Guide and Tutorial for Networked Parallel Computing, MIT Press, Cambridge, 1994.

- [2] Griffioen J., Yavatkar R., Finkel R., Unify: A Scalable, Loosely-Coupled, Distributed Shared Memory Multicomputer, *University of Kentucky, Computer Science Dept. Technical Report*, Ocak 1993.
- [3] Manchek B., *Heterogeneous Distributed Computing*, 1995.
- [4] Manchek R., “Workstation Solutions: Opening the Door to Heterogeneous Network Supercomputing”, *Supercomputing Review* Volume 4, Number 9, Utah Department of Mathematics - University of Utah, USA, September, 1991, pp 44-46.
- [5] Özmen A, Durmuş B, “Dağıtık Sistemlerde Paralel Performans ve Karmaşıklık Analizi”, 1. Ulusal Yüksek Performanslı Bilişim Sempozyumu, Gebze-Kocaeli, 24-25 Ekim 2002, pp 45-49.

PARALLEL VIRTUAL MACHINE (PVM)

B. DURMUŞ & A. ÖZMEN

Abstract. Some applications require high performance computing environment. These applications can be real-time control systems, soft real-time simulators or long running non-interactive compute bound tasks. In this paper, a parallel virtual machine (PVM) and its graphical interface (XPVM) have been introduced to meet high performance computing demands. PVM is a software package designed as a library extension to an operating system. A virtual parallel system can easily be build from networked computers using PVM. These computers can also be heterogeneous according to their architecture and operating systems. Therefore, PVM makes it possible to build affordable high performance computers for universities and research companies using ordinary workstations and personal computers.

Keywords: Distributed computing, host , PVM, XPVM

* Dumlupınar Üniversitesi Mühendislik Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü, Kütahya, Türkiye bdurmus@dumlupinar.edu.tr

* Dumlupınar Üniversitesi Mühendislik Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü, Kütahya, Türkiye ozmen@dumlupinar.edu.tr