

ATM Anahtarlarının Performansının İyileştirilmesi

Abdülhamit Subasi

Elektrik ve Elektronik Mühendisliği Bölümü

Kahramanmaraş Sütçü İmam Üniversitesi

asubasi@ksu.edu.tr

ÖZET

Geniş bantlı ağların temelini oluşturan ATM anahtarları veri iletişiminin gereksinimi olan gigabit/sn seviyesindeki iletim hızlarını desteklemektedir. Böyle yüksek seviyedeki anahtarlama hızını sağlamak için ATM ağlarında kullanılan anahtar elemanlarının daha verimli bir şekilde çalışması için çeşitli çalışmalar yapılmıştır. Bu çalışmada anahtarlama elemanlarının girişlerinde bulunan bellek modülleri birleştirilerek, anahtar girişleri tarafından ortak olarak kullanılacak esnek bir yapı ortaya konulmuştur. ATM ağlarının performansı anahtarlama elemanlarında bulunan tampon belleklerin tasarım şekillerine ve saat mekanizmalarının verimli bir şekilde tasarlanmasına bağlıdır. Büyük çevrim, küçük çevrim ve sık (smart) çevrim olmak üzere üç çeşit saat çevrimi önerilmiştir. Çıkış tampon bellek ve alındı (acknowledge) yöntemini kullanan sık çevrim saat modeli performans olarak diğerlerinden daha iyidir. Bunun yanında, bu modelin dezavantajı, çok yüksek trafik yükü altında paketlerin sırasının bozulması veya kaybolması ihtimali ile karşı karşıya bulunmasıdır. Bu çalışmada, ATM anahtar modüllerinin girişlerindeki tampon bellek paylaşımı bir bellek havuzu oluşturularak, sık çevrim saat modelinin daha verimli bir yapıya kavuşturulması sağlanmıştır.

1. GİRİŞ

ATM'de veri transferinin paketler (cells) halinde yapılması nedeniyle, farklı bant genişliğine sahip çeşitli servisleri tek bir ağda toplamak mümkündür. ATM'nin tipik özellikleri; çok hızlı bağlantı, basitleştirilmiş protokol ve yüksek kapasiteli anahtarlama sistemine sahip olması şeklinde sıralanabilir. Bu sistemlerde elde edilmesi gereken maksimum **üretilebilir hız (throughput)** ancak anahtarlama noktalarının dağıtılması ve paralel olmasıyla sağlanabilmektedir [1].

Çeşitli çalışmalarda, ATM anahtarlarının analiz ve simülasyon çalışmaları yapılmıştır. ATM anahtarlama elemanlarında, tampon bellek tasarımının yanında saat tasarımı da büyük önem arz etmektedir. Saat periyodunun tasarımı özellikle ağın birim zamanda gönderebileceği toplam mesaj miktarı olarak değerlendirilen üretilebilir hız miktarını etkilemektedir. Yoon ve arkadaşlarının yaptığı çalışmada [2], saat çevriminin denetim sinyalini en son kademedeki ilk kademeye varması için yeterli bir zaman miktarını kapsadığı varsayılmaktadır. Ding ve Bhuyan [3] bu varsayımın dezavantajını göz önünde bulundurarak

küçük çevrim modelini önermiştir. Bunun yanında önerilen bu modelde yüksek trafik yükü altındaki ağda çevrim kayıplarına sebep olduğu gözlemlenmiştir. Bhogavilli ve Abu-Amara [4], sık çevrim (smart cycle) olarak adlandırılan yeni bir model önermiştir. Bu modelde saat mekanizması çıkış tampon bellek ve alındı kullanılarak çevrim kayıplarını önlemeyi amaçlamıştır. Önerilen bu model her bir anahtar girişindeki tampon belleklerin yani sıra her bir çıkışta da ikiser tampon bellek bulunmasını önererek, daha küçük saat çevrimi kullanarak daha iyi sonuçları ortaya koymuştur. Bunun yanında bu modelde yüksek ağ trafığında paketlerin sıralarının değişmesi veya kaybolması ihtimali ortaya çıkmıştır. Bu çalışmada sık çevrim modeli kullanan sistem için dinamik giriş tampon bellek önerilerek ATM ağlarının performansının iyileştirilmesi amaçlanmıştır. Statik giriş tampon belleklerin kullanıldığı sık çevrim modelinde, asiri trafik yükü altındaki ağda, paketlerin sıralamasının değişmesi veya kaybolması dinamik giriş tampon bellek kullanılarak önlenmiş; bunun yanında ağdaki üretilen mesaj miktarı artmış ve paketlerin gecikmesi ise azalmıştır. Bu çalışmada önerilen bu model simülasyon programı yardımıyla modellenerek statik ve dinamik giriş tampon bellekli ağların karşılaştırılması yapılmıştır.

2. ATM AĞLARI

ATM, B-ISDN servisleri için geliştirilmiş bir anahtarlama ağıdır. ATM çok geniş spektrumdaki iletişim servislerini sağlamak için ideal bir yapıdır. Bu yapıda kullanıcılar arasındaki veri iletişimi, ATM hücreleri olarak adlandırılan sabit uzunluktaki paketler vasıtasıyla sağlanır [5], [6], [7], [8].

ATM, LAN veya WAN'lar üzerinden ses, görüntü, video ve veri aktarmak için geniş bantlı bir hücre aktarım (cell relay) teknolojisidir. Veriler, bölümlenmiş paket verilerini taşıyan sabit boyda çerçevelerden meydana gelen hücrelere yerleştirilirler. Hücreler küçük olduğundan, ağ anahtarları üzerinden aktarılmalari da kolaydır [9], [10].

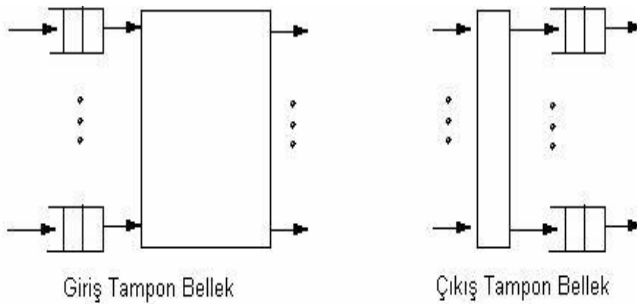
ATM'deki anahtar tasarımı, veri iletişim hızı ve hat kapasitesi tüm sistemin performansını artıracak bir şekilde tasarlanmıştır. ATM anahtarlarının diğer anahtarlardan farkı; hat üzerinde oluşan veri iletişim hızının çok fazla (50 Mbps – 4 Gbps) olmasıdır [11], [12], [13].

Donanım ve yazılım alternatiflerinin çok fazla miktarda olması nedeniyle çok farklı yapıda ATM anahtarlama sistemleri tasarlamak mümkündür. ATM anahtarlarının kolayca yenilenebilmesine (upgrade) olanak tanımak için, yazılımın donanıma bağlı ve donanımdan bağımsız olmak üzere iki farklı yapıda oluşturularak sistemin modüler olmasının sağlanması gerekmektedir [14], [15].

Ethernet, Fiber Distributed Data Interface (FDDI) ve Token Ring gibi geleneksel ağlar, herhangi bir anda sadece bir tek düğümün iletim yapabildiği paylaşımlı ortamı kullanırlar. ATM ağlarında ise, düğümler aynı anda iletim yapabilir ve istenen sayıda bağlantı kurulabilir. Pek çok düğümden gelen bilgi, bir bit akışı halinde çoklanır. Bu sistemde, ATM anahtarı bir genel servis sağlayıcıda bulunabileceği gibi, kurumun iç ağının bir parçası da olabilir [16].

ATM ağı içerisinde herhangi bir anahtarın girişlerinde bulunan paketlerden birden fazlası aynı çıkışa gitmek istediğinde, çıkışta bir çekişme (contention) meydana gelir. Bu durumda paketlerden birisi gitmek istediği çıkışa gönderilirken, diğerleri tampon belleklerde belirli bir süre bekletilir. Tasarımda bu gereklilik göz önünde bulundurularak anahtarlara belirli miktarda tampon bellek yerleştirmek gerekir. Tamponların nasıl yerleştirileceği konusundaki temel yaklaşım Şekil 1'de sematik olarak gösterilmektedir.

Giriş tampon bellek kullanılması yönteminde, anahtar elemanının her bir girişi için ayrı bir tampon birimi bulunur. Bu yapıda, paketler arasındaki çekişme (contention) anahtarlama elemanının giriş biriminde çözülür. Yani bütün paketler anahtar girişlerine geldiklerinde bu giriş kuyruklarından birisine girer ve daha sonra anahtar elemanının kontrol birimi tarafından gerekli çıkışa iletilerek varacağı yere gönderilmesini sağlar. Giriş tampon bellekte, veri paketlerinin hareketleri FIFO (ilk giren ilk çıkar) yöntemiyle sağlanır. Bu tampon bellek yapısında satırbaşı (HOL-Head of Line) problemi yaşanır. HOL probleminde farklı iki giriş tampon bellekte bulunan paketler aynı çıkış adresine gitmek istediklerinde paketlerden birisi çıkışa gönderilirken, diğeri diğer paket ise tamponlarda bekletilir. Bekletilen paket bir sonraki saat çevriminde çıkışa gönderilir.



Şekil 1. Farklı tampon bellek yaklaşımları

Giriş Tampon belleğin tersine olarak, çıkış tampon bellekte anahtar elemanının her bir çıkışıyla ilgili bir tampon bellek kuyruğu bulunur. Paketler çıkış adreslerine göre çıkışta bulunan bellek paket kuyruklarından birisine gönderilir. Bu yapıda paketler arasındaki çekişme (contention) anahtarlama elemanının çıkış biriminde çözülür.

3. TAMPON BELLEKLİ ATM AĞLARI

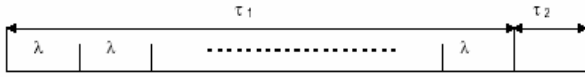
ATM'de ağı özelliğinin yanında saat çevriminin tasarımı da ağı performansını önemli ölçüde etkilemektedir. Yoon ve arkadaşlarının yaptığı çalışmada [2] saat çevrimi iki aşamadan oluşmaktadır. t_1 olarak adlandırılan birinci aşamada her bir kademede denetim mekanizması iki işlem yapmaktadır: bir sonraki kademede boş tampon bellek olup olmadığı kontrol edilir ve önceki kademe bu durumdan haberdar edilir. t_2 olarak adlandırılan ikinci kademede ise, eğer sonraki kademeye gidecek beklemekte olan paket varsa ve sonraki kademede boş tampon bellek varsa kontrol devresi paketin bir sonraki kademeye ilerlemesini sağlar. Paketin gideceği kademede tampon belleğin dolu olması durumunda ise kontrol devresinin bir sonraki saat çevriminde dolu olan tampon belleğin boşalmasını denetlemesi gerekir. Bunun için saat çevriminin, kontrol sinyalinin son kademeden ilk kademeye kadar yayılabilmesini sağlayacak büyüklükte olması gerekmektedir. Eğer anahtar elemanları n kademeden oluşmuş ise ve her bir kademe için n kadarlık bir süre gerekli ise $t_2 = n \cdot n$ olacaktır. Şekil 2 (a)

Büyük çevrim modelinin dezavantajları Ding ve Bhuyan [3] tarafından ortaya çıkarılarak bunun yerine küçük çevrim modeli önerilmiştir. Şekil 2(b). Önerilen bu küçük çevrim modelinde i kademesindeki paket sadece $i+1$ kademesindeki tampon belleğin boş olması durumunda bir sonraki kademeye gidebilmekte, aksi durumda ise beklemektedir. Bundan dolayı $t_1 = ?$ olmakta ve ağ boyutuna göre değişmemektedir. Bunun yanında küçük çevrim modelinde bütün kademelerdeki tamponların dolu olması durumunda, ki bu en kötü durum olmaktadır, n tane küçük saat çevriminin kaybolmasına yol açmaktadır.

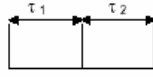
Bhogavilli and Abu-Amara [4] sık çevrim olarak adlandırılan daha verimli bir model önermişlerdir. Önerilen bu modelde kontrol ve veriler t_2 anında paralel olarak iletilmekte, böylece t_1 gerek kalmamaktadır. Bu modelde kontrol devresi birbirinden bağımsız üç durumlu makine gibi çalışmakta olup, paket gönderimi, kabul sinyalinin (ACK) alınması ve kabul edilen paketler için kabul sinyalinin gönderilmesi işlemleri birlikte yapılmaktadır. Her saat çevriminin başında, hangi veri paketinin bir sonraki kademede anahtarlama elemanının tampon belleğinin boş veya dolu olması durumuna bakılmaksızın ilerleyeceği kontrol devresi tarafından belirlenir. Bundan dolayı k kademesindeki veri paketi, $k+1$ kademesindeki anahtar elemanının giriş tampon belleğinin boş olması, veya $k+1$ kademesindeki anahtarlama elemanındaki

tampon belleğin dolu olması durumunda ise, veri paketinin $k+2$ kademesindeki giriş tampon belleğe gidebilmesi durumunda, k kademesinin çıkış tampon belleğine gitmektedir. Bundan dolayı paketler kontrol sinyalini kullanmadan yalnızca bir çevrimde ilerlemekte ve böylece daha iyi bir throughput ortaya çıkmaktadır. Bunun yanında eğer gönderilen paket $k+1$ kademesindeki anahtar elemanı tarafından kabul edilmezse, ve paket yeniden gönderilmezse paket kaybolmaktadır. Bundan dolayı, Bhogavilli ve Abu-Amara her bir anahtarlama elemanının çıkış tampon belleğinin uzunluğunun iki olması gerektiğini göstermişlerdir.

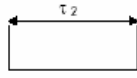
Çıkış tampon belleğinin her ikisinin de dolu olması durumunda, her zaman kabul edilmeyen bir paket bulunmakta ve bu paketin kaybolmaması için yeniden gönderilmesi gerekmektedir. Bu durumda bir saat çevriminin bosa gittiği bir durum ortaya çıkmaktadır. Bu çalışmada bu durumu önlemek için, anahtar girişlerindeki bellekler bir araya toplanarak bir tampon bellek havuzu oluşturulmuş ve bu yeni modele dinamik tampon bellekli ağ denilmiştir. Anahtarlama elemanlarındaki tampon bellekleri dinamik yapmak suretiyle, ağın başarımının arttığı bu çalışmada gösterilmiştir.



(a) Büyük Çevrim



(b) Küçük Çevrim



(c) Şık Çevrim

Sekil 2. Büyük, küçük ve sık çevrim saat modelleri

Tampon bellekli ağların performansını değerlendirmek ve karşılaştırmak için iki önemli performans ölçüsü, üretilen is (throughput) ve gecikmedir (delay). Üretilen is, her bir ağ çevriminde geçen ortalama paket sayısıdır. Gecikme ise, her bir paketin ağdan geçmesi için gerekli süredir (ortalama saat çevrimi sayısı) [2].

Bu ölçülerin değerleri, ağın dış ortamı ile olduğu gibi ağ içerisindeki çeşitli ağ parametrelerine de bağlıdır. Bu yüzden, o paketleri üreten ve kabul eden ortam hakkında bazı basitleştiren varsayımlar yapmak ve ağ ortamında bazı kısıtlamalar yapmak gerekir. Bundan dolayı, bu ağ parametrelerini dış çevreden bağımsız olarak hesaplamak için bazı varsayımlar yapılarak **tekdüze**

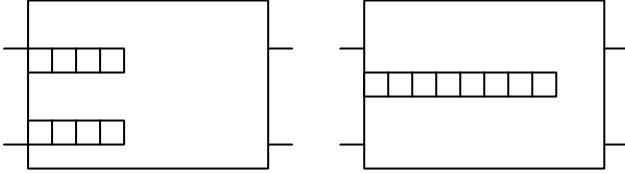
(**uniform**) trafik denilen bir model uygulanmıştır [14], [15]. Tekdüze trafikte:

- 1) Paketler N tane ağ girişi için N tane birbirinden bağımsız süreç olarak üretilirler. Yani N tane işlemci, her kademe çevriminde $q(1)$ giriş yükü olasılığı ile paket üretirler.
- 2) Her bir girişe aynı trafik yükü uygulanmaktadır. Tekdüze trafiği benzetmek için her kademe çevriminde, her bir paketin varacağı adres rasgele belirlenmektedir
- 3) Paketlerin varış adresleri tüm ağ çıkışları için tekdüze olarak dağıtılmıştır. Bunun anlamı, her bir anahtar tamponu aynı anda eşit sayıda pakete sahiptir.
- 4) Tampon bellekteki paketler ilk giren ilk çıkar (FIFO) şeklinde hareket ettirilmektedirler.
- 5) Her bir anahtar içerisindeki çelişmeler (conflicts) rasgele çözülürler.
- 6) Ağ eşzamanlı bir şekilde çalışmaktadır. Yani paketler belirli zaman aralıklarında ve bu zaman aralıklarının başlangıcında gönderilmekte olup, bu zaman aralıkları kesikli (discrete) zaman aralıklarıdır. Bu durum ağ içerisindeki paketlerin sabit uzunlukta ve zaman dilimi içerisine sığacak büyüklükte olduğunu göstermektedir.
- 7) Ağ içerisinde hiçbir şekilde hiçbir paket kaybolmamaktadır. Bunun anlamı, herhangi bir paket ancak gitmek istediği anahtar girişindeki tampon belleğin boş olması ve kendisini kabul etmesi gerektiği anlamına gelmektedir. Aksi durumda paket bir sonraki anahtar girişindeki tampon belleğe gitmeyip, bir sonraki saat çevrimini bekleyecektir.
- 8) Paketler çıkışa vardığında ağdan hemen ayrılırlar. Ağın çıkışında herhangi bir bekleme söz konusu olmamaktadır. Paketler çıkışa geldiklerinde hemen çevre birimi tarafından kabul edilmektedirler. Yani çevre birimleri en az anahtar çıkışlarıyla aynı hızda olmak zorundadırlar.

Paketler ağın giriş kapılarında bağımsız, tekdüze, rasgele işlemler tarafından üretildiği varsayımı, ağ başarımının üst sınırlarını hesaplamak ve analizi kolaylaştırmak için yapılmaktadır. Paketleri üreten modüller, diğer modüllerdeki paketlerin üretimine bağlı ise, ağ içindeki paket aktarımının güncel hızı, modüller bağımsız olduğunda gözlenen aktarma oranından daha az olacaktır. Buna ek olarak, eğer varış adreslerinin dağılımı, tekdüze olmazsa, ağın bazı kısımları daha çok tıkanacak ve sonuçta üretilen iste de bir azalma gözlenecektir. Benzer biçimde, paketler çıkış kapılarına ulaştığında en kısa zamanda ağdan ayrıldığı varsayımı, paketlerin gideceği ortamın arabaglayıcıdan daha hızlı olması gerektiği anlamına gelir. Buraya kadar yapılan bütün varsayımlardan çıkarılacak sonuç, ağdaki bütün anahtarlama elemanlarındaki (Switching Elements-SEs) paket dağılımı aynı ve istatistiksel olarak bağımsız olduğu anlamına gelir.

Tampon bellekli ağlarda ise, her bir anahtarlama elemanının girişlerinde belirli bir miktar tampon bellek bulunur. Eğer paket çelişmesi ortaya çıkarsa, çelişen paketin biri rasgele seçilip gönderilir, diğer paketler ise tampon belleklerde bekletilirler. Bu çalışmada, burada söz

konusu olan yapı, “**sabit veya statik tampon bellekli ag**” olarak adlandırılmaktadır. Statik tampon bellekli ağlarda her bir anahtarlama elemanının, Sekil 3 (a)’da gösterildiği gibi belirli bir miktar tampon bellek bulunmaktadır. Her bir anahtarlama elemanındaki tampon bellekler Sekil 3 (b)’deki gibi bir havuzda birleştirilerek, ortak bir tampon bellek meydana getirilmektedir. Burada sözü edilen yeni sistem ise “**dinamik tampon bellekli ag**” olarak adlandırılmaktadır.



a) Statik Tampon Bellek b) Dinamik Tampon Bellek

Sekil 3. Statik ve dinamik tampon belleklerin gösterimi

Ding ve Bhuyan [3] küçük saat çevrimli bir model geliştirerek, YLL modeline benzer bir analiz yöntemiyle sistem başarımını iyileştirdiğini göstermişlerdir [2].

Bhogavilli ve Abu-Amara [4] saat çevrimi kaybını azaltan yeni bir model önermişlerdir. Bu çalışmada çıkış tamponları ve onaylamaları kullanmak suretiyle, bir sik çevrim (smart cycle) modelinin [3]’deki küçük çevrim modeline göre üstün bir ag başarımı ortaya koyduğu gösterilmiştir. Çıkış tampon modüllerinde iki paketi depolamak için tampon belleğin eklenmesi, gerekli olan üstün ag performansını sağlamakta yeterli olduğu gösterilmiştir. Bu modelle ilgili kapsamlı analitik model [4]’te geliştirilmiş ve benzetim (simülasyon) çalışmalarıyla doğruluğu gösterilmiştir.

Sik çevrim modelinde, giristeki trafik yükünün yoğunluğuna bağlı olarak, paketleri sırasız bir şekilde çıkışa gönderebileceğinden dolayı bu paketlerin çıkışta tekrar sıraya konulmaları gerekir. Giriş yükü sabit bir degerden daha büyük olduğunda akıllı veya küçük saat çevrimli ağlar, tampon taşmasından (buffer overflow) dolayı paketleri kaybedebilirler [3], [4].

4. SIMÜLASYON ve MODELLEME

Bu çalışmada farklı yapıdaki ATM anahtarlarının simülasyonu yapılmıştır. Önerilen yeni model ATM anahtarını oluşturma için gerekli donanıma sahip olmadığımızdan dolayı sistemin bilgisayarda benzetimi yapılarak sonuçlar elde edilmiştir. Sistemi modellemek için C++ programlama dili kullanılmıştır. Program içerisinde tampon belleklerin benzetimi kuyruk yapısı kullanılarak oluşturulmuştur. Benzetim çalışmalarının tümünde ag boyutu (N), anahtar girişi (a), anahtar çıkışı (b), kademe sayısı (n), tampon bellek boyutu (m) ve giriş yükü (q(1))

giriş parametreleri; üretilen is (S) ve gecikme (d) ise çıkış parametreleri olarak kullanılmıştır.

Benzeticinin tasarımında bazı varsayımlar yapılmış olup bunlar aşağıda sıralanmıştır.

- 1) Paketler N tane ag girişi için N tane birbirinden bağımsız süreç olarak üretilirler. Yani N tane işlemci, her kademe çevriminde q(1) giriş yükü olasılığı ile paket üretirler.
- 2) Her bir girişe aynı trafik yükü uygulanmaktadır. Tekdüze trafik benzetimini gerçekleştirmek için, her kademe çevriminde işlemciler tarafından üretilen paketlerin varis yerleri rasgele sayı üretici (random number generator) tarafından istatistiksel olarak bağımsız üretilmektedirler.
- 3) Tampon bellekteki paketler ilk giren ilk çıkar (FIFO) şeklinde hareket ettirilmektedirler.
- 4) Her bir anahtar içerisindeki çelişmeler (conflicts) rasgele çözülürler.
- 5) Ag eşzamanlı bir şekilde çalışmaktadır. Yani paketler belirli zaman aralıklarında ve bu zaman aralıklarının başlangıcında gönderilmekte olup, bu zaman aralıkları kesikli (discrete) zaman aralıklarıdır. Bu durum ag içerisindeki paketlerin sabit uzunlukta ve zaman dilimi içerisinde sigacak büyüklükte olduğunu göstermektedir.
- 6) Ag içerisinde hiçbir şekilde hiçbir paket kaybolmamaktadır. Bunun anlamı, herhangi bir paket ancak gitmek istediği anahtar girişindeki tampon belleğin boş olması ve kendisini kabul etmesi gerektiği anlamına gelmektedir. Aksi durumda paket bir sonraki anahtar girişindeki tampon belleğe gitmeyip, bir sonraki saat çevrimini beklemektedir.
- 7) Paketler çıkışa vardığında ağdan hemen ayrılırlar. Ağın çıkışında herhangi bir bekleme söz konusu olmaz. Paketler çıkışa geldiklerinde hemen çevre birimi tarafından kabul edildiklerinden, çevre birimleri en az anahtar çıkışlarıyla aynı hızda olmak zorundadırlar.
- 8) Is çıkarma yeteneği ve gecikme, ağın her bir çıkışındaki veri paketleri, ag boyutu ve benzetimde harcanan zamana bölünerek ortalama deger olarak hesaplanmaktadır.

Simülasyon programı çalıştırıldığında ağın boyutu (N), kademe sayısı (n), statik tampon boyutu (ms), dinamik tampon boyutu (md), anahtar boyutu (a) sisteme girilmektedir. Simülasyonda paketler N tane girişten rasgele rand fonksiyonu kullanılarak üretilmektedir. Eger ilk kademedeki anahtarlama elemanının tampon belleğinde boş yer varsa, üretilen paket tampon belleğe gönderilmekte; boş yer yoksa paket bir sonraki kademe çevrimini beklemektedir. Bu şekilde ilk kademedeki anahtarların her bir girişi için ayrı ayrı birer kuyruk oluşturulmaktadır.

Daha sonra paketler üzerindeki varis adres bilgilerindeki her bir kademe ile ilgili degerler kullanılarak ag içerisindeki çeşitli kademelerden geçerek çıkış portuna

ulasmaktadırlar. Her kademe çevriminde, gecikmeyi hesaplamada kullanılan toplam paket zamanı her bir paket için birer artırılmaktadır. Çıkış portunda üretilen is ve gecikmeyi hesaplamak için paketler sayılmaktadır. Üretilen is ve gecikme değerlerinin kararlı duruma erisebilmesi için simülasyon programı içerisinde t zaman değeri birer artırılmakta; kararlı duruma erildiğinde simülasyon programı durdurulmaktadır. Kararlı durum, hesaplanan S (üretilen is) ve d (gecikme) değerlerinin bir önceki kademe çevriminde hesaplanan değerlerle arasındaki farkı önceden belirlenen bir ϵ sayı değerine erdiğinde oluşmaktadır. Buradaki t zaman değeri ağ boyutuna bağlı olarak 150 000 ila 200 000 değerleri arasında değişmektedir.

$d = \text{toplam paket zamanı} / \text{toplam paket} / n$ (kademe sayısı).

$S = \text{toplam paket} / t$ (toplam kademe çevrimi) / N (ağ boyutu).

5. SONUÇLAR

ATM ağlarında, paketlerin varis adreslerinin rasgele doğasından dolayı tamponların bazılarının tam dolu, bazılarının ise tamamen dolu olmadığı görülmüştür. Yani herhangi bir anda bir anahtarlama elemanının bir girişindeki tamponlar tamamen dolu iken, diğer girişinde boş tampon olduğu görülebilir. Buradan herhangi bir anahtarlama elemanının boş olan tamponu, dolu olan diğer giriş tarafından kullanılabilmesi sonucuna varılabilir. Yani anahtarlama elemanları içerisinde bulunan tamponlar birden fazla anahtar girişinin kullanabileceği şekilde tasarlanarak, ağın başarımı artırılabilir. Buradan yola çıkarak tampon modülleri Şekil 3'te görüldüğü gibi birleştirip tampon havuzu oluşturulabilir. Bu çalışmada oluşturulan bu yeni model "**dinamik tampon bellekli ağ**" olarak adlandırılmaktadır. Bu şekilde oluşturulan dinamik tamponlu ağ, gecikmeyi (delay) azaltmak ve üretilen is miktarını (throughput) artırmak suretiyle sistemin performansını artırmaktadır.

Bu teoremin sağlanması, 4. bölümde detaylı bir şekilde anlatılan bilgisayar simülasyonu kullanılarak yapılmıştır. Bu simülasyon programında, ağ boyutu (N), kademe sayısı (n), tampon sayısı (m), giriş yükü (q) ve zaman (t) girdileri; is çıkarma yeteneği (S) ve gecikme (d) ise çıktıları oluşturmaktadır. Simülasyon programı çalıştırıldığında N tane ağ girişinden paketler rasgele üretilmekte; daha sonra paketler ağ içerisinde geçerek çıkışta sayılmakta, sonuçta üretilen is miktarı ve gecikme hesaplanmaktadır.

Bu çalışmada statik ve dinamik tamponlu ağların karşılaştırılmasında aynı simülasyon programı kullanılmıştır. Kullanılan program sistemin dinamik ve statik tampon bellekli olmasına bağlı olarak, farklı iki tampon bellek modelinin kuyruk yapısının benzetimini yapan farklı altyordamlar kullanılmıştır. Benzetim çalışması

sonucunda elde edilen tüm karşılaştırmalı çizimlerde, statik ve dinamik tamponlu sistemlerdeki her bir anahtar içerisindeki tampon sayısı eşit alınmıştır. Bunun sonucunda statik ve dinamik tampon bellekli ağlar birbirleriyle karşılaştırılırken kullanılan çizimlerde, her bir anahtar girişindeki tampon sayıları eşit olarak ele alınmıştır.

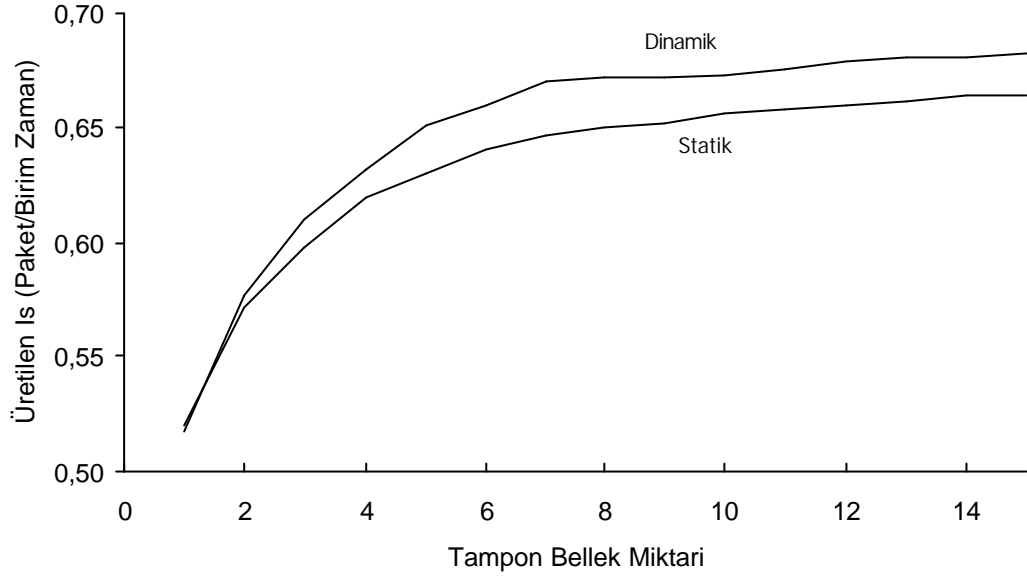
Ağ boyutu $n=3$ ve anahtarlama elemanı boyutu $a=2$ olan bir ağ için tampon sayısına karşılık üretilen is miktarı Şekil 4'te, gecikme ise Şekil 5'te çizilerek dinamik ve statik tamponlu ağlar karşılaştırılmıştır. Bu şekillerde, tamponları dinamik yapmak suretiyle ağ performansının arttığı görülmektedir. Yani, dinamik tamponlu ağda üretilen is miktarı statik tamponlu ağdan fazla, buna karşın gecikmesi daha azdır.

Şekil 6'da ağ boyutu $n=5$ için, tampon bellek miktarına karşın üretilen is miktarının değişimi çizilmiştir. Şekil 7'de ağ boyutu $n=5$ için, tampon bellek miktarına karşın gecikme miktarının değişimi çizilmiştir.

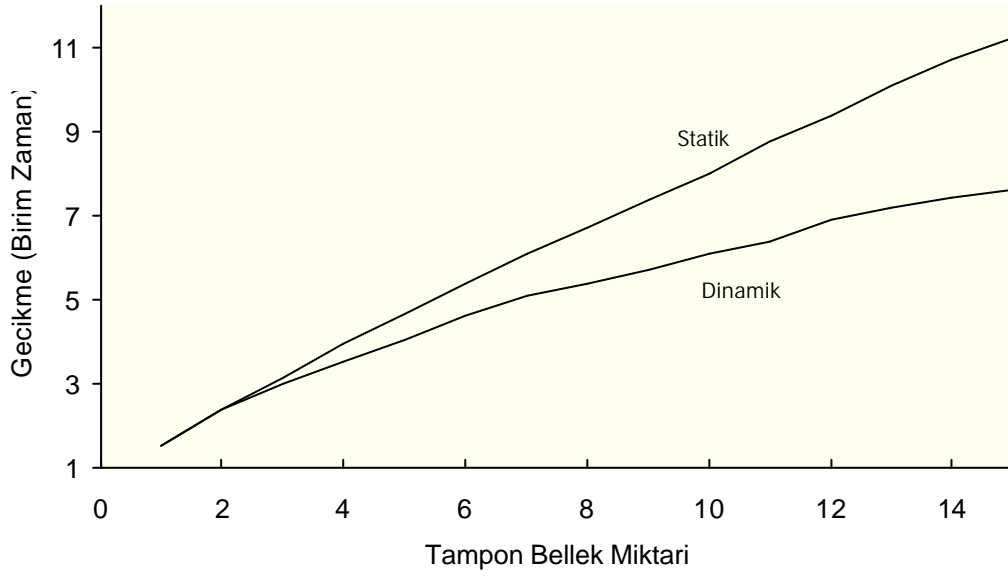
Şekil 8'de tampon bellek boyutu 6 için, ağ boyutuna miktarına karşın üretilen is miktarının değişimi çizilmiştir. Şekil 9'da tampon bellek boyutu 6 için, ağ boyutuna karşın gecikme miktarının değişimi çizilmiştir. Bu şekillerde görüldüğü gibi, ağ boyutunun büyümesiyle sistemin ürettiği is ve gecikme miktarı giderek azalmaktadır. Statik ve dinamik tampon belleklerin karşılaştırılması göz önüne alındığında; dinamik tampon bellekli ağlarda, statik tampon bellekli ağlardan daha fazla is üretildiği, bunun yanında daha az gecikme oluştuğunu ortaya koymaktadır.

Bu şekillerde de görüldüğü gibi, tamponları dinamik yapmak suretiyle ağın performansı önerilen yeni modellerle belirli bir miktarda artırılmaktadır. Yani, dinamik tamponlu ağlar statik tamponlu ağlara kıyaslandığında üretilen is miktarının daha fazla, gecikmenin ise daha az olduğu gözlenmektedir. Çünkü dinamik tamponlu ağda tampon bellekler tüm girişler için ortak kullanıldığından, yani anahtar elemanı içerisinde bir tampon bellek havuzu oluşturulduğundan, boş durumda bulunan bellek herhangi bir anahtar girişindeki paket tarafından kullanılabilen, böylece ağdaki paketlerin gecikmesi azalmakta, aynı zamanda üretilen is miktarı artmaktadır.

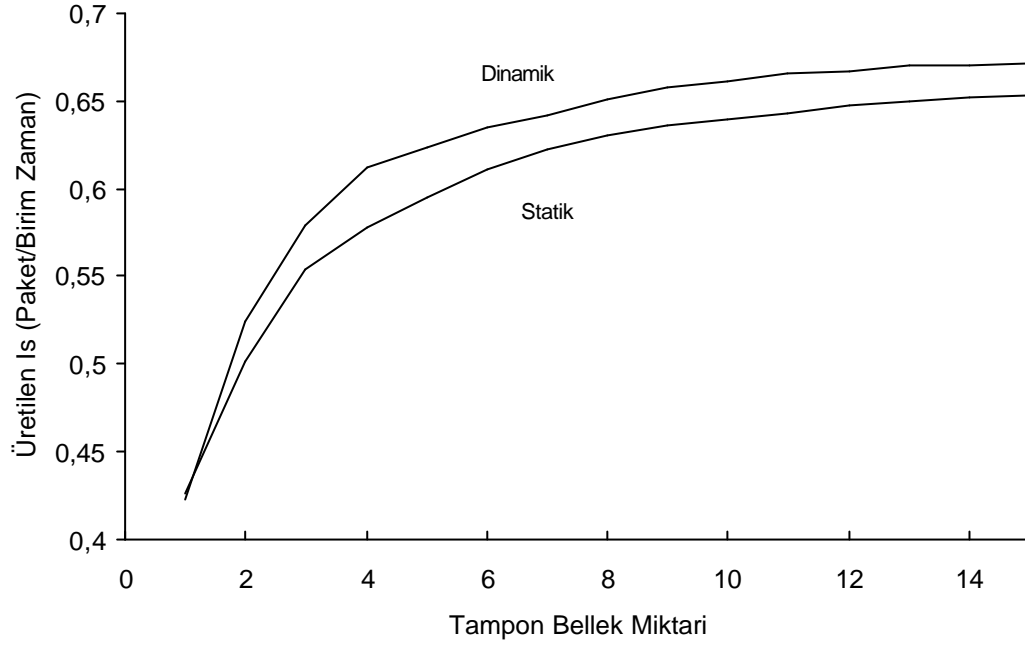
Buradan elde edeceğimiz sonuç; tampon bellekleri dinamik yapmak suretiyle ortak bir bellek havuzu oluşturulmakta ve böylece ağın performansının arttığı açıkça görülmektedir. Bu da bize, önerilen yeni sistemin daha iyi bir performans sağladığını ortaya koymaktadır.



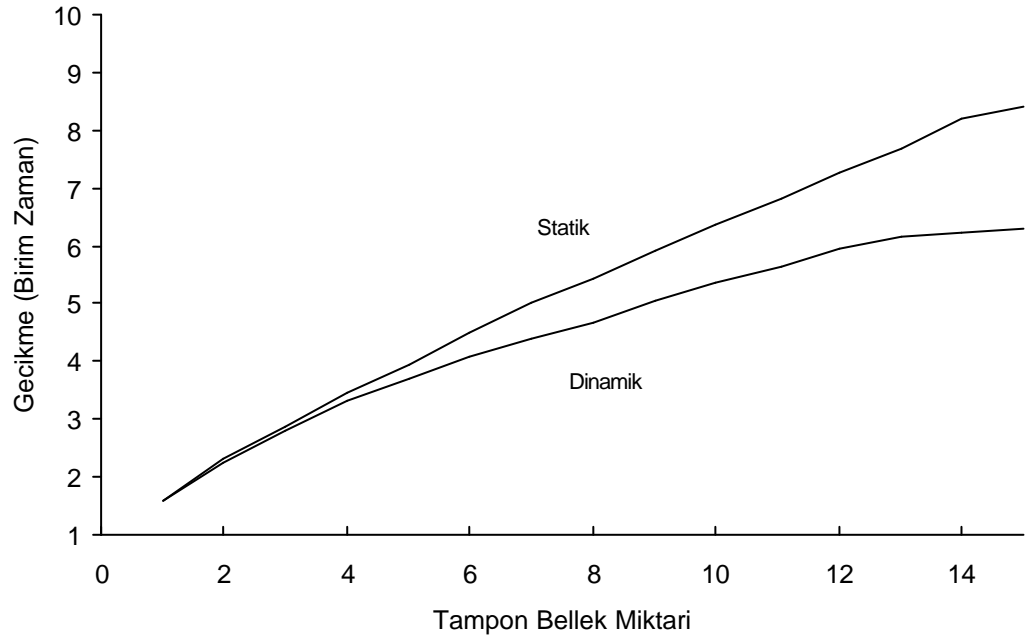
Sekil 4. Statik ve dinamik tampon belleklerin karsilastirilmesi ($a=2, n=3, q(1)=1$)



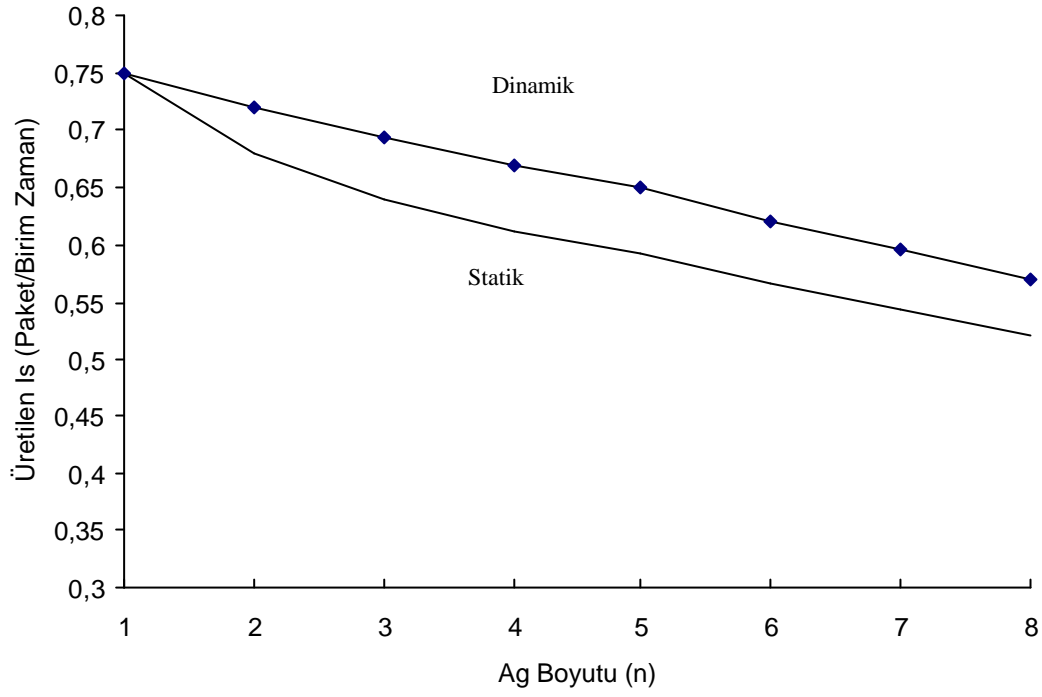
Sekil 5. Statik ve dinamik tampon belleklerin karsilastirilmesi ($a=2, n=3, q(1)=1$)



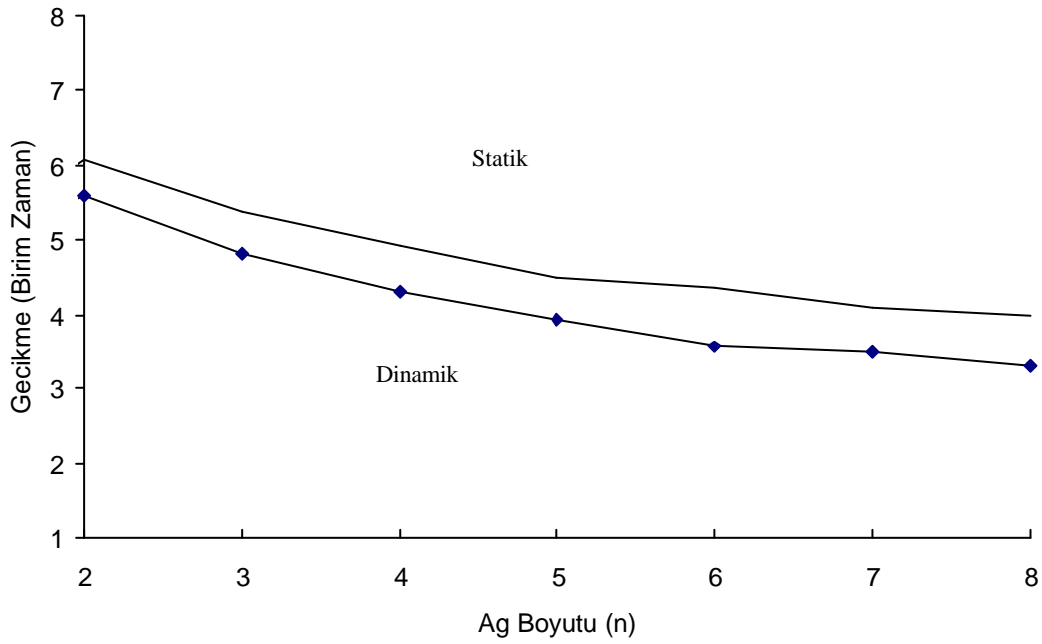
Sekil 6. Statik ve dinamik tampon belleklerin karşılaştırılması ($a=2$, $n=5$, $q(1)=1$)



Sekil 7. Statik ve dinamik tampon belleklerin karşılaştırılması ($a=2$, $n=5$, $q(1)=1$)



Sekil 8. Statik ve dinamik tampon belleklerin karşılaştırılması ($a=2$, $m=6$, $q(1)=1$)



Sekil 9. Statik ve dinamik tampon belleklerin karşılaştırılması ($a=2$, $m=6$, $q(1)=1$)

6. TARTISMA VE ÖNERİLER

ATM ađlari, bilgisayar ađlari nin olu sturulmasında bir devrim yaratabilecek bir veri ile tim teknigidir. Yerel ve geniş ađlarda güvenilir olan bu teknoloji, yüksek hızda veri transferleri sağlamaktadır. Bu veriler bilgi, faks, gerçek-zamanlı video, CD kalitesinde ses ve görüntüleme sek illeri olarak ortaya çı kmaktadır.

Bu çalı smanın ana hedefi, ATM ađ performansını artıracak bir anahtar modeli ortaya koymaktır. Bu çalı smada çok tamponlu ađlar için yeni bir model geliştirilmiş olup; bu yeni model **dinamik tampon bellekli** ađ olarak adlandırılmıştır. Daha önce önerilen ađlar ise **statik tampon bellekli** ađ diye adlandırılmıştır.

Bu çalı smasının sonucunda, dinamik tamponlu ađların performansının statik tamponlu ađlardan daha iyi olduğu gösterilmiştir. Yani, dinamik tamponlu ađda üretilen iş miktarı statik tamponlu ađdan fazla, bunun yanında gecikme miktarı ise daha azdır.

Statik ve dinamik tamponlu ađlardaki performans artışı ađ boyutundaki artış ile daha da belirginleşmektedir. Yani ađ boyutunun artmasıyla, dinamik tamponlu ađlarda üretilen iş miktarı artarken gecikme miktarı statik tamponlu ađlara nazaran daha da azalmaktadır. Bu da bize daha büyük boyuttaki ađlar için dinamik tamponlu bellek kullanımının daha da avantajlı olacağını göstermektedir. Sonuç olarak bu tezde yapılan çalı smada ortaya konulan yeni model, günümüzde yaygın bir şekilde kullanılan ATM ađları için daha iyi bir performans sağlamaktadır. Uygulama açısından bakıldığında ise sistemin daha da karmaşık olacağı ve bunun sonucunda da maliyetinin artacağı göz önünde bulundurulması gereken bir konu olmaktadır.

7. KAYNAKLAR

- [1] Theimer, T. H., Rathgeb, E. P., and Huber, M. N., "Performance Analysis of Buffered Banyan Networks," IEEE Trans. On Comm., pp. 269, Feb. 1991.
- [2] Yoon, H., Lee, K. Y., and Liu, M. T., "Performance Analysis of Multibuffered Packet-Switching Networks in Multiprocessor Systems," IEEE Trans. Comput., 39 (3), pp. 319, March 1990.
- [3] Ding, J. and Bhuyan, L. N., "Finite Buffer Analysis of Multistage Interconnection Networks," IEEE Trans. On Comput. Vol. 43, No. 2, pp. 243, Feb. 1994.
- [4] Bhogavilli, S. K. and Abu-Amara, H., "Design and Analysis of High Performance Multistage Interconnection Networks," IEEE Trans. On Comput. Vol. 46, No. 1, pp. 110-117, Jan. 1997.
- [5] Kumar, B., "Broadband Communications: A Professional's Guide to Frame Relay, SMDS, SONET and B-ISDN", New York: Mc-Graw Hill, 1995.
- [6] Robertazzi, T. G., "Performance evaluation of high

speed switching fabrics and networks : ATM, broadband ISDN, and MAN technology", New York : IEEE Press, 1993.

[7] Chen, T. M. and Stephen, S. L., "ATM Switching Systems", Artech House, Incorporated, 1995.

[8] Chris, D., Konangi V. K., and Sreetharan M., "Broadband switching : architectures, protocols, design, and analysis", Los Alamitos, Calif. : IEEE Computer Society Press, 1991.

[9] Flood, J. E., "Telecommunications Switching, Traffic and Networks", Prentice Hall, 1995.

[10] Spohn, D. L., "Data network design: Packet Switching Frame Relay 802.6 - DQDB SMDS ATM B-ISDN, SONET," New York : McGraw-Hill, 1993.

[11] Onvural, R. O., "Asynchronous transfer mode networks : performance issues", Boston : Artech House, 1994.

[12] Black, U. D., "ATM: Foundation for Broadband Networks", Prentice Hall, Inc., 1995.

[13] Konakondla, S. K., "Design, simulation and performance analysis of a shared memory ATM switch for B-ISDN networks", Cleveland State University, 1994.

[14] Awdeh, R.Y. and Mouftah, H.T., "Design and performance analysis of input-output buffering delta-based ATM switch with backpressure mechanism", IEE Proceedings: Communications v 141 n 4, pp 255-264, Aug 1994.

[15] Huang, T. Y. and Wu, J. L. C., "Performance analysis of ATM switches using priority schemes", IEE Proceedings: Communications v 141 n 4, p 248-254, Aug 1994.

[16] Schmidt, A. and Campbell, R., "Internet Protocol Traffic Analysis with Applications for ATM Switch Design", Computer Communication Review Vol: 23 Iss: 2 p. 39-52, April 1993.