Emrehan HALICI

DAHA AZ YERDE DAHA ÇOK VERİ

Katkısı bulunanlar :
Ferhat BÜYÜKKÖKTEN, Oğuz IŞIKLI

Bilgisayar dünyası, özellikle son yıllarda büyük miktarlarda verinin kullanılmasına, depo edilmesine ve transfer edilmesine sahne olmaktadır. Bu gelişme ise birtakım ekonomik problemleri beraberinde getirmektedir.

Örnek olarak, bir kuruluş veritabanı sistemi, zamanla daha çok veriyi kapsamı içinde tutmak zorundadır. Bu durum ise ek disk ve manyetik teyp ünitelerinin kullanıma sokulmasını gerektirmektedir. Veritabanı sistemindeki gelişmenin bir diğer yönü ise kullanıcı sayısındaki artıştır. Farklı noktalar arasında, büyük ölçekli bilgi transferini sağlayabilmek için, modem ve multiplexer gibi yan cihazlar sisteme dahil edilmektedir.

Yukarıda anlatılan sorunların hepsi, esasında daha çok depolama ünitesi ve iletişim olanağı sağlanarak çözülebilir. Ancak, günün ihtiyaçları paralelinde hızla artan veri miktarı, bu çözümü ekonomik açıdan caydırıcı hale getirmektedir. Düşünülebilecek bir başka yöntem ise, eldeki mevcut verinin daha etkili bir kod ile daha az yerde temsil edilebilmesi ve gerektiğinde aynen orijinal haline çevrilmesidir. Dikkatle incelendiği zaman pek çok tür bilgi, orjinalinden daha küçük boyutlarda depolanabilecek özellikler içermektedir. Bu yüzdendir ki, veri sıkıştırma, günümüz bilgisayar teknikleri içinde en önemlilerinden biridir.

İki Türlü Yaklaşım...

Veriyi daha az yerde depolayabilmek için, iki farklı yaklaşım kullanılabilir:

A) Sistem daha tasarım halinde iken, mümkün olduğu kadar az yer kaplayan bir veri yapısının kullanılmasına çalışılabilir. Verinin iki yerde tanımlandığı alanlar önlenebileceği gibi, kalan kısımların mantıklı bir şekilde temsil edilmesi, bir veritabanı tasarımının önemli noktalarındandır.

Bir örnek olarak, bir tarihi belirtmek için 5 Ekim 1989 yerine sayısal olarak 05 10 1989 ifadesi depolanabilir. Daha da ileri giderek gün değeri 31'i geçemeyeceğinden bu sayıyı ifade edecek 5 bit, aynı şekilde ay değeri için 4 bit ve yıl değeri için ise 1900-2027 aralığındaki 127 yılı ifade edebilecek 7 bit ile bütün tarih 16 bitten oluşan 2 bayt dahilinde temsil edilebilir.

Dikkatle gerçekleştirilen bir tasarımın sağlayabileceği tasarruf etkili olmakla birlikte, zamanla sistemde meydana gelebilecek değişiklikler tasarımdan hiç etkilenmeyebilir. İşte bu yüzden daha dinamik ve veri yapısına uygun bir yaklaşım gereklidir.

B) Transfer edilecek veya depolanacak verinin yapısal özelliklerini dikkate alacak bir kodlama sistemi, orjinal veriyi çok daha küçük boyutlara indirgeyebilir. Bazı karakter veya karakter dizilerinin diğerlerinden fazla kullanılması gibi istatistik özellikler göz önüne alındığında, bu karakter ve dizilerin daha küçük kodlarla ifade edilmesi paralelinde, bir birim bilgi daha az yere toplanabilir. Bu ikinci yaklaşım çerçevesinde verinin yapısına uygun pek çok sıkıştırma tekniği uygulanabilir.

Sıkıştırmanın Getirdiği Faydalar...

- Öncelikle programların çalışma sürelerinde bir kısalma söz konusudur. Esasında veriyi kodlamak ve tekrar orjinal haline getirmek için extra bir zaman harcansa da, disk gibi yan bir hafıza ünitesinden veri alırken harcanan zaman, çok daha fazla azalacağından, genel toplamda zamandan tasarruf edilebilir.

- Veri iletişiminin maliyeti, genelde iletişimin süresi ile doğru orantılıdır. Sıkıştırma ile daha az bir yerde depolanan verinin transfer maliyeti bu aranda az olacaktır.

- Aynı şekilde sıkıştırılmış verinin transferinde meydana gelebilecek hatalar azalacaktır. Çünkü hata olasılığı sabit kaldığı halde, yollanan veri miktarı azalmıştır. Bir yan avantaj ise, sıkıştırma sonucu oluşturulan yeni kodun bir tür güvenlik seviyesi sağlamasıdır.

Bazı Teknikler...

- Yukarıda bahsedildiği üzere, depolanacak veya transfer edilecek veride birtakım sembollerin arka arkaya tekrar edilmesi gibi yapısal özellikler bulunabilir. "Run-length coding" diye bilinen bir teknik ile bu durumun avantajlı bir hale getirilmesi mümkündür. Yöntem, tekrar edilen sembol dizilerinin sadece sembol ve tekrar sayısı ile belirtilmesi esasına dayanmaktadır. Örnek olarak,

aaaaabcddddddee serisi
Xa5bcXd7ee şeklinde kodlanabilir.

Burada X sıkıştırma uygulandığını belirten özel bir semboldür ve veride arka arkaya 5 tane a olduğunu belirtmektedir. Görüldüğü üzere bu yöntemin etkili olabilmesi için, 4 veya daha fazla tekrarlı sembol dizileri gerekmektedir.

- Bazı verilerde birtakım sembol gruplarının daha çok kullanılması doğaldır. Örneğin BASIC program dosyalarında Print, GOTO, INPUT ve REM gibi komutlara daha çok rastlanmaktadır. Bu noktadan hareket ederek, işlenecek verideki böyle özelliklere sahip sembol gruplarının özel ve daha kısa kodlarla belirtilmesi, programın genel boyutunu küçültecektir.

- Oldukça etkili bir başka yöntem ise özelliğini, kullandığı istatistik metottan almaktadır. Sözgelimi dilin yapısına bağlı olarak, metin dosyalarında bazı harflere daha fazla rastlamak mümkündür (İngilizce'de e harfinde olduğu gibi). Bu gibi karakterlerin belirlenerek, daha az sayıda bit ile temsil edilmesi sayesinde, bir karakter için harcanan ortalama bit sayısı düşürülebilir ve bir metin dosyasının sıkıştırılmasında çok tatmin edici sonuçlar elde edilebilir. Bu sistem ile çalışan Huffman coding ve benzeri tekniklerle 8 bitlik kodların ortalama 2,5 bitlik alanlarda ifadesi mümkün olabilmektedir.

Bir Örnek... Faksimile Uygulamaları

Günümüz iletişim teknolojisinde oldukça önemli bir pay sahibi olan telefax sistemleri, veri sıkıştırma tekniklerinden bir veya birkaçını kullanmaktadırlar. Öncelikle bu sistemlerin çalışma prensiplerini kısaca belirtelim.

Bilindiği üzere sayısal faksimileler, girdi olarak verilen belirli boylardaki kâğıtlar üzerinde bulunan yazılı veya basılı bilgileri, elektronik olarak algılayıp sayısal sinyallere çeviren, bunları bir hat ile yollayabilen ve aynı şekilde gelen sinyalleri değerlendirerek tekrar kâğıt üstünde kullanıcıya sunabilen sistemlerdir. Kâğıttaki bilginin transfer edilebilecek hale gelmesi, kâğıt üstündeki her noktanın açıklık-koyuluk derecesinin sayısal olarak ifadesi ile mümkündür. Standardlara göre 21,5 x 33 cm'lik bir sayfanın elektronik sinyal haline getirilmesi, 1730'ar resim elemanından (pixel) oluşan 850 satırın fotoelektrik olarak incelenmesi ile gerçekleştirilmektedir. Yaklaşık 1 milyon bitten oluşan bu bilgi ise 4800 bps ile çalışan bir transfer sisteminde yaklaşık 3,5 dakikada işlenebilir.

Ancak bu resim elemanlarının büyük bir kısmının beyaz olduğu düşünülürse, satırların hepsinin ifadesi yerine sadece satırlar arası göreceli farkların ifadesinin daha az yer tutacağı anlaşılabilir.

Örneğin

.....00001110011010.....ve
.....00011110001100.....satırları arasında
.....---x-----x-xx-.....şeklinde belirlenebilecek fark

ifadeleri bahsedilen "run-length" tekniği ile 1730 elementten çok daha az bir yere sıkıştırılabilir. Uygulanacak metot, sırayla birbirini takip eden ikişer satır arasındaki farkları kodlamak ve iletişim kanalına bu kodu yollamaktır. Alıcı faksimile ise, bu farkları kullanarak 850 satırı oluşturabilir. Bu tip yöntemlerle 1/10 oranında veri sıkıştırılması sağlanabilmekte ve bu doğrultuda, oldukça yüklü rakamlar oluşturan telekomünikasyon maliyetleri düşürülebilmektedir.

Bütün bu tekniklerin amacı daha kolay, daha ucuz bilgi iletişimidir. İletişim olgusunun hayatımızda gittikçe daha önemli roller oynamaya başladığı bu asırda, dikkatle hazırlanmış, yazılım ve donanım uyumu sağlanmış başarılı veri sıkıştırma tekniklerinin zamanla daha değerli bir hale geleceği kanısındayız.

---

7 nolu ödüllü sorumuzun doğru cevabı 5777 ve 5993'tür.

Kurada kazanan okurlarımız:

1- Zafer Ulusoy

Kozağaç Mah. 266. Sok. No: 60/3 Buca/İZMİR

2- Erkan Can

Ziya Gökalp Cad. No: 23/2 Kızılay/ANKARA

3- Oğuz Konak

Temsan A.Ş. P.K. 224 Diyarbakır.

Bilgisayar dergisi, kurada kazanan okurlarımızı, dergiye 1 yıl süreyle ücretsiz abone yapacağını duyurmuştur. Kendilerine teşekkür ediyoruz.

ÖDÜLLÜ SORU NO: 8

Halkayı koparmadan, sayılı pulları oynatarak öyle üç parçaya ayırın ki, ilk iki parçanın çarpımı üçüncü parçayı versin.

(Örnek: 6 ve 3, 4'ün yanına getirilerek, şu üç parça elde edilebilir: 28,907 ve 15463. Fakat ne yazık ki, ilk iki sayının çarpımı üçüncüyü vermiyor.)