

KASTAMONU ÜNİVERSİTESİ

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

ÖDEV RAPORU

Veri İletişimi ve Bilgisayar Ağları

Ödev 3 (Üç)

184410029

RECEP POLAT

İçindekiler

Bölüm 1. **Hata Sezme 3**

* 1. Hata Sezme Nedir? 3

1.1.1. Hata Algılama Nedir? 3

1.1.2. Hata Düzeltme Nedir? 3

1.1.3. Hata Algılama ve Düzeltme Teknikleri Nelerdir? 3

Bölüm 2. **Hata Algılama ve Hata Düzeltme Teknikleri 4**

2.1. CRC Algoritması 4

2.2 CheckSum Algoritması 4

2.3 LRC ve VRC Algoritması 5

2.4 Hamming Algoritması 5

Kaynakça 6

Bölüm 1

**Hata Sezme**

* 1. **Hata Sezme Nedir?**

Paket

ALICI

GÖNDERİCİ

Onay

Elektronik ortamda veri aktarımları her zaman ideal şekilde gerçekleşmez. Cihazlar ve kablolar birçok çevresel faktöre bağlıdır. Haberleşme esnasında veri iletimi sırasında veri katarlarında bozulmalar meydana gelebilir. Mesela;

* Onay gidebilir.
* Onay zarar görebilir.
* Onay hiç gitmemiş olabilir.

Bunlar ilk olarak hata algılama (Error Detection) dediğimiz tekniklerle belirlenirler. Ve burada herhangi bir hata düzeltme yapılmaz. Sadece hata belirlenir. Hata olduğunu ve bu hatanın belirlendiğini düşünecek olursak, önümüzde iki seçenek vardır;

* Alıcı vericiye bir mesaj iletir. Bu mesaj verinin tekrar gönderilmesi sağlar ama bu verinin orijinal hali gelene kadar sürecektir.
* Alıcı bu mesajı kendi bildiği teknik ve kod ile çözümleyerek hangi bitin hatalı olduğunu tespit eder ve düzeltir.

Kısaca bahsedecek olursak Hata Algılama ve Hata Düzeltme nedir?

* + 1. **Hata Algılama:** Alıcıya gönderilen verinin herhangi bir faktörün neden olduğu bozulmaların algılanması işlemidir.
    2. **Hata Düzeltme:** Alıcının gönderilen verideki oluşan bozulmaları gidererek orijinal veriyi elde etme işlemidir.

Hataları Algılamada ve algılanan Hataları Düzeltmede birden fazla teknik vardır. Peki, bu teknikler nelerdir? Bu tekniklerin Hata Algılama ve Hata Düzeltmedeki sınırlılıkları nelerdir? Performansları ne kadar iyidir? Her hatayı bulabilirler mi? Bunları inceleyeceğiz.

* + 1. İlk olarak bu teknikler nelerdir?

Bu sorunun cevabı aşağıdaki gibidir;

* CRC Algoritması
* CheckSum Algoritması
* LRC ve VRC Algoritması
* Hamming Algoritması

Bölüm 2

**Hata Algılama ve Hata Düzeltme Teknikleri**

Bu bölümde bazı hata algılama ve düzeltme tekniklerini inceleyeceğiz.

2.1. **CRC** **Algoritması**

İngilizce Cyclic Redundancy Check olarak bilinen bu algoritma, hata sezildiği durumlarda bozulmanın onarılmasını sağlar. Bu teknik yaygın olarak kullanılır ve hata saptama oranı en fazla olan algoritmadır.

Peki, nasıl çalışır?

Bu algoritmada veri katarı bir dizi olarak ele alınır. Yöntem temel olarak;

* Çerçeveyi gönderen bilgisayar tarafından çerçevenin içeriğine göre ve çerçevenin sonuna eklenecek şekilde bir denetim seti üretir.
* Eklenmiş halini alıcıya iletir.
* Her iki tarafın da (gönderici, alıcı) bildiği bir kod ile bu veri bölme işlemine tabi tutulur.
* Eğer kalan var ise bitlerde hata oluşmuş demektir ve alıcı çerçevenin tekrar gönderilmesi ister.
* Eğer kalan yok ise bitlerde hata oluşmamış demektir.

Denetim seti(üreteç polinom), iletişim ortamının ve beklenen iletim hatalarının türüne uyacak şekilde seçilir. Bu seçim çok önemlidir çünkü denetim seti hangi tür hataları fark edileceğini tanımlar.

CRC Algoritması ile aşağıdaki hataları belirleyebiliriz;

* Tüm tek bit hatalarını
* Tüm çift bit hatalarını
* Tüm tek sayılı bit hatalarını
* R’den küçük tüm hata patlamalarını (R: Üreteç polinomun derecesidir.)
* R’den büyük çok fazla hata patlamalarını

2.2 **CheckSum Algoritması**

Bu algoritma CRC ile aynı işlevi sağlar ama CRC algoritması kadar kesin koruma yöntemi sağlayamaz. Bu algoritmanın temeli oldukça basittir;

* Gönderilen tüm kelimeleri(word) topla
* Elde ettiğin toplamı sağlama toplamı olarak gönder.
* Alıcı da aynı işlemleri tekrar ederek bulduğu sağlama toplamını gönderilenle karşılaştırır.
* Bir hata durumunda sonuçlar farklı çıkacağından alıcıyı iletimde bir hata olduğu konusunda uyarır.
* Örn, IP 16 bit uzunluğunda bir sağlama toplamı yapar.

Checksum Algoritması tek sayıda bit içeren tüm hataları algılar. Ancak her hatalı veri için hatalıdır mesajı vermez. Çünkü bir segmentin veya bir veya daha fazla biti ahsar görürse ve ikinci bir segmentteki karşılık gelen bit veyaz zıt değer bitleri de hasar görürse bu sütunların toplamları değişmez ve alıcı bir sorun algılamaz. En ünlü CheckSum algoritması “Internet CheckSum” algoritmasıdır.

2.3 **LRC-VRC Algoritmaları**

Düşey artıklık denetleme (VRC), bir karakterde iletim hatası meydana gelip gelmediğini bulmak üzere eşlik kullanan bir hata bulma tekniğidir. Bu nedenle, VRC’ye bazen **karakter eşliği** de denir. VRC’de, iletimden önce her karaktere bir eşlik biti eklenir. VRC’de çift ya da tek eşlik kullanılabilir.

Bir mesajda iletim hatası meydana gelip gelmediğini belirlemek için eşlik kullanan bir hata bulma tekniğidir; bu nedenle bazen mesaj eşliği adını alır. LRC’de, her bit konumunun bir eşlik biti vardır. Başka bir deyişle, mesajdaki bir karakterin b0’ı, mesajdaki öteki karakterlerin hepsinin b0’ı ile Mutlak Veya’lanır. Benzeri şekilde b1, mesajdaki öteki karakterlerin hepsinin b1’iyle Mutlak Veya’lanır ve süreç bu şekilde devam eder. Temel olarak LRC, bir mesajı oluşturan karakterlerin Mutlak Veya’lanmasının sonucudur; buna karşın VRC, tek bir karakterdeki bitlerin Mutlak Veya’lanmasıdır. LRC’de yalnızca çift eşlik kullanılır.

Verileri göndermeden önce LRC bit sırası vericide hesaplanır, daha sonra sanki mesajın son karakteriymiş gibi iletilir. Alıcıda, LRC alman verilerden tekrar hesaplanır ve tekrar hesaplanan LRC, mesajla iletilen LRC’yle karşılaştırılır. Eğer iki LRC aynı ise, iletim hatasının gerçekleşmediği varsayılır. Eğer LRC’ler farklıysa, bir iletim hatası meydana gelmiş demektir.

LRC bütün iletim hatalarının % 95 ila % 98’ini bulur. LRC, aynı bit konumunda çift sayıda karakter hatalıysa, iletim hatası bulmaz.

VRC ve LRC aynı anda kullanılırsa, hatanın algılanmadığı tek durum şu olur: çift sayıda karakterde çift sayıda bit hatalı olduğunda ve bu iki karakterde aynı bit konumları hatalı olduğunda; böyle bir şeyin meydana gelme olasılığı çok düşüktür.

2.4 **Hamming Algoritması ve Kodlaması**

Kodlama teknolojisinde veri bitleri ile veriye eklenen hata saptama bitlerini birleştiren birleşik ileti birimi kod sözcük(codework) olarak adlandırılır.

İki kod sözcük arasında kaç bit konumunun değiştiğini bulmak için önce iki kod sözcüğe XOR operatörü uygulanır ve sonra bu işlem sonucunda elde edilen 1’lerin sayısı toplanır.

Bir kod sözcükteki tek bir bit bozulursa geçersiz bir kod sözcük ortaya çıkacağından bütün tek bitli hataları saptayabilir. Aksi takdirde, tek başlarına VRC ve LRC, yalnızca bir hatanın meydana geldiğini bulabilirler.

2.4.1. **(7,4) Hamming Kodlaması**

Bu Hamming Kodlama yönteminde her 4 bitlik data için 3 bit kontrol biti eklenir. Geliştirilen bu algoritma herhangi bir tek bitlik hatayı düzeltebilir ya da bütün tek veya iki bitlik hataları tespit edebilir. Eğer iletim sırasında tek bitlik bir hata meydana gelmişse bunu matris yardımı ile bozulan bitin sırasını bildirir.

**KAYNAKÇA**

Balıkesir Üniversitesi Yüksek Lisans Ders Notu: <http://w3.balikesir.edu.tr/~myuksek/dersnotu/veri.pdf>

Hata Saptama ve Düzeltme Teknikleri: <https://www.slideshare.net/ercanbulus/bilgisayar-alar-ders-notlar-62399050>

Kemal AKYOL | Veri İletişimi ve Bilgisayar Ağları Ders içi Ders Notları

Bilgisayar Ağları ve İnternet Mühendisliği | Rıfat ÇÖLKESEN