



دانشگاه شهید مدنی آذربایجان

دانشگاه شهید مدنی آذربایجان
دانشکده فناوری اطلاعات و مهندسی کامپیوتر
دکتر محسن حیدریان

گروه مهندسی کامپیوتر
نیمسال اول ۱۴۰۳

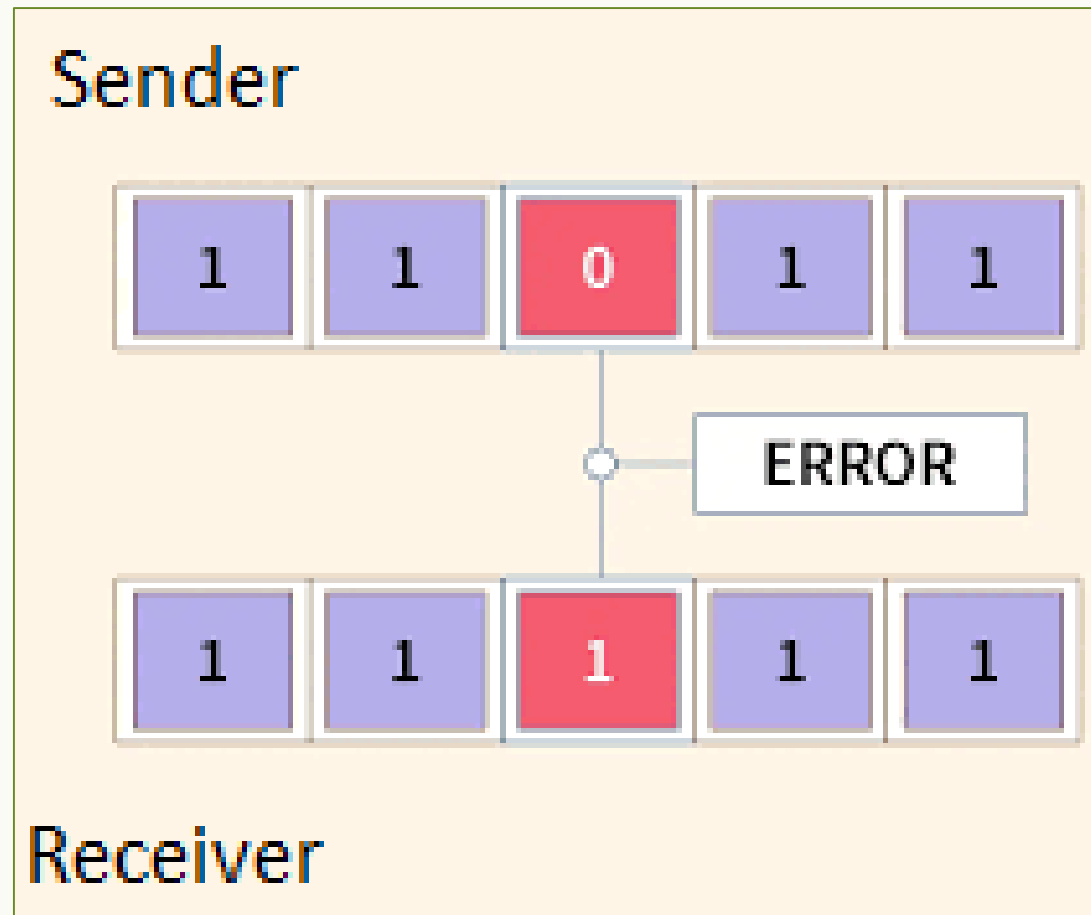
درس

Computer Networks

شبکه های کامپیوتری

دانشگاه شهید مدنی آذربایجان - دکتر محسن حیدریان

Error



عناوین تدریس

تعریف انواع خطای بیتی

فاصله همینگ

کد همینگ

کد همینگ ۱۲ بیتی

قوانین کد همینگ ۱۲ بیتی

مثال

انواع خطای پیتی در داده های ارسال شده

برای آنکه بتوانیم یک کلمه Word از داده ها را بگونه ای کد گذاری کنیم که قابلیت تشخیص و تصحیح خطا را داشته باشد، باید تعدادی بیت به آن اضافه کنیم. اگر طول یک Data Word به اندازه D بیت باشد، آنگاه پس از کد گذاری یک کلمه کد شده Codeword به اندازه C بیت خواهد بود: بگونه ای که $C > D$ می باشد. لذا تعداد حالت هایی که این کدها تولید می کند، بیشتر خواهد شد ولی تمام این حالت ها درست نیستند، و این همان چیزی است که باعث می شود سیستم بتواند وجود خطا را تشخیص دهد. یعنی اگر یک بیت در یکی از این حالات غیرمجاز یا غیر قابل قبول باشد، باشد، سیستم می فهمد که خطایی روی داده است. در بعضی از روش ها، سیستم در یک سری از حالات می تواند خطای بوجود آمده را نیز اصلاح کند. روش ارائه شده باید این قابلیت را داشته باشد که از بین C بیت موجود D بیت اصلی را خارج کند. به این عمل اصطلاحاً Decoding می گویند. یکی از مشکلات استفاده از کدینگ این است که سیستم مجبور است تا یک مدت زمانی را صرف عملیات Encoding و Decoding کند که باعث ایجاد سربار Overhead در سیستم می شود.

کدهای Hamming برای کشف و اصلاح خطا

در دهه ۱۹۵۰ میلادی ریچارد همینگ که در آزمایشگاههای شرکت بل کار می کرد به معرفی دسته ای از کدهای اصلاح کننده خطا پرداخت که بنام خود او کدهای همینگ خوانده می شوند. شاید ساده ترین روش برای آشکار کردن خطای یک بیت در یک بایت، استفاده از بیت توازن است. **فاصله همینگ بین دو Codeword برابر است با تعداد بیت هایی که آنها با هم متفاوتند.** یعنی نشان میدهد که اگر در اثر خطا یک کد بخواهد به یک کد دیگر تبدیل شود، چند بیت از آن باید تغییر کند تا این تبدیل انجام شود بدون آنکه سیستم آن را خطا به حساب آور. در تئوری اطلاعات فاصله همینگ بین دو رشته برابر طول تعداد مکانهایی است که سمبولهای متناظر متفاوت هستند. به معنای دیگر، کمترین تعداد جایگزینی هایی است که یک رشته به یک رشته دیگر تغییر پیدا کند، یا تعداد خطاهایی که یک رشته به رشته دیگر تبدیل گردد. چند مثال برای فاصله همینگ بین چند رشته:

Toned و roses فاصله همینگ سه هست.

۱۰۱۱۱۰۱ و ۱۰۰۱۰۰۱ فاصله همینگ دو هست.

۲۱۷۳۸۹۶ و ۲۲۳۳۷۹۶ فاصله همینگ سه هست.

کدهای ۱۰۱ و ۰۱۱ در ۲ بیت با یک دیگر متفاوت هستند. در نتیجه فاصله همینگ بین آنها برابر ۲ است. اما کدهای ۱۰۱ و ۱۰۰ فقط در یک بیت با هم تفاوت دارند. در نتیجه اگر یک خطا در بیت کم ارزش آنها روی دهد، یکی از آنها را به دیگری تبدیل می کند و سیستم متوجه وجود خطا نخواهد شد. فاصله همینگ به اندازه ۲ تضمین می کند که اگر یک خطای تک بیتی اتفاق بیفتد سیستم حتما متوجه بروز خطا خواهد شد.

کدهای Hamming برای کشف و اصلاح خطا

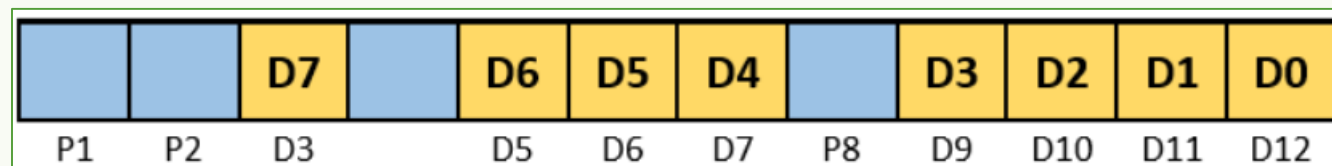
فاصله کد برابر است با کمترین فاصله همینگ که بین هر دو کد موجود در یک مجموعه کد وجود دارد. یعنی اگر مثلاً در یک روش کدینگ فاصله کد برابر ۲ باشد به این معنی است که هیچ کدام از کدها با کدهای دیگر فاصله همینگ کمتر از ۲ ندارند. برای مثال مجموعه کدهای $\{001, 010, 100, 111\}$ همگی باهم فاصله ۲ دارند. در نتیجه این کد می تواند هر خطای تک بیتی را تشخیص دهد.

به عنوان مثالی دیگر کدهای $\{000, 111\}$ دارای فاصله ۳ هستند پس می توانند هر خطای تک بیتی یا دو بیتی را تشخیص دهند. اما اگر فرض شود احتمال خطای دو بیتی کم است، این کد را می توان به عنوان روشی که می تواند خطاهای تک بیتی را اصلاح Correct کند، نیز استفاده شود.

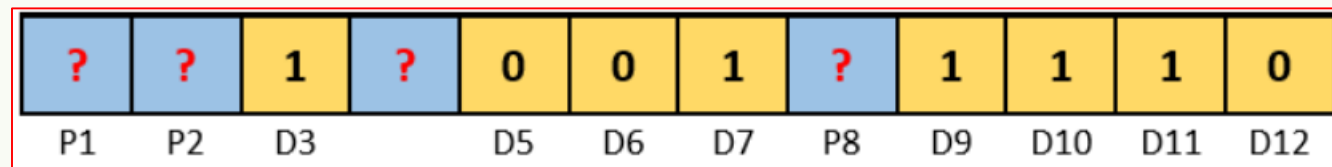
به عنوان یک تعریف ریاضی می توان گفت : برای آنکه بتوانیم تا حداکثر t بیت خطا را تشخیص دهیم، نیاز به حداقل فاصله کد به اندازه $t+1$ داریم. ولی برای آنکه بتوانیم تا حداکثر t بیت خطا را تصحیح کنیم، نیاز به حداقل فاصله کد $2t+1$ داریم.

کدهای Hamming برای کشف و اصلاح خطا

فرض کنید که قصد داریم داده‌ای را از یک فرستنده به گیرنده‌ای ارسال کنیم. داده ما ۸ بیت طول دارد. برای تصحیح خطا در داده‌های ۸ بیتی بنا به نظریه‌ی فاصله‌ی همینگ، نیاز به ۴ بیت داده‌ی تصحیح خطا خواهیم داشت. پس در مجموع برای ارسال ۸ بیت داده با استفاده از همینگ کد، نیاز به ۱۲ بیت خواهیم داشت.



همان‌طور که در تصویر فوق مشاهده می‌کنید، محل قرارگیری بیت‌های داده در ساختار همینگ کد پشت سرهم نیست؛ مثلاً آخرین بیت از داده‌ی ارسالی D7 باید در موقعیت سوم قرار بگیرد. خانه‌های مشخص‌شده با رنگ زرد، محل قرارگیری داده‌ها هستند و خانه‌ی مشخص‌شده با رنگ آبی محل قرارگیری بیت‌های توازن هستند. قبلاً اشاره شد که برای ارسال‌های ۸ بیتی، نیاز به ۴ بیت توازن داریم که با نام‌های P1, P2, P4, P8 مشخص شده‌اند. فرض کنید قصد داریم داده‌ی 0x9E را ارسال کنیم. معادل باینری آن ۱۰۰۱۱۱۱۰ می‌شود. با جای گذاری آن، تصویری مطابق شکل زیر خواهیم داشت:



کدهای Hamming برای کشف و اصلاح خطا

حال باید بیت‌های توازن را محاسبه و جای گذاری کنیم. هر بیت توازن در واقع توازن زوج از بیت‌های ۱ است.

0	0	1	1	0	0	1	1	1	1	1	0
P1	P2	D3		D5	D6	D7	P8	D9	D10	D11	D12

$$P1 = D3 \oplus D5 \oplus D7 \oplus D9 \oplus D11$$

$$P2 = D3 \oplus D6 \oplus D7 \oplus D10 \oplus D11$$

$$P4 = D5 \oplus D6 \oplus D7 \oplus D12$$

$$P8 = D9 \oplus D10 \oplus D11 \oplus D12$$

به‌عنوان نمونه بیت توازن P1، توازن زوج از بیت‌های ۳ و ۵ و ۷ و ۹ و ۱۱ است. تعداد مقدار ۱ در این چهار بیت، عدد زوج ۴ است (تنها بیت ۵ صفر است)؛ بنابراین P1 باید صفر باشد. مابقی بیت‌ها را نیز مطابق فرمول ارائه‌شده در عکس محاسبه می‌کنیم که مقدار P2 یک، مقدار P4 یک و مقدار P8 نیز صفر می‌شود. با جای گذاری داده‌های موردنیاز، داده ارسال می‌شود و در گیرنده دریافت می‌شود؛ اما به دلیل نویزهای محیطی، داده با خطا و تغییر بیت D10 از ۱ به صفر دریافت شده است.

0	0	1	1	0	0	1	1	1	0	1	0
P1	P2	D3		D5	D6	D7	P8	D9	D10	D11	D12

کدهای Hamming برای کشف و اصلاح خطا

گیرنده شروع به محاسبه‌ی بیت‌های توازن می‌کند. با توجه به تغییر وضعیت D10 و با توجه به تأثیر D10 بر روی P2 و P8، این دو بیت توازن دارای مقداری صحیح نخواهند بود درحالی‌که دو بیت توازن P1 و P4 دارای مقداری صحیح هستند.

برای پیدا کردن بیتی که دچار خطا شده، کافی است که ارزش بیتی P2 و P8 را با هم جمع کنیم؛ یعنی $2+8$ که می‌شود بیت دهم. همان‌طور که مشاهده کردید به راحتی علاوه بر تشخیص خطا، قادر خواهیم بود خطای ایجادشده را تصحیح نیز کنیم. همین‌کد کاربردهای فراوانی از جمله در ارتباط‌های رادیویی و شبکه‌ای دارد. این الگوریتم باعث کاهش خطا و رفع آن می‌شود.

