Дәріс 8. Хэмминг коды. Негізгі идеялары, минималды қашықтықты есептер және қателерді түзету. Іске асыру мысалдары.

Дәріс жоспары.

1. Хэмминг коды.
2. Негізгі идеялары, минималды қашықтықты есептер және қателерді түзету.
3. Іске асыру мысалдары.

Хэмминг коды.

Хэмминг коды - қателерді анықтау және түзетуге арналған ең кең тараған артық кодтау әдістерінің бірі. Оны 1950 жылы американдық ғалым Ричард Хэмминг ұсынған. Бұл кодтың негізгі мақсаты – деректерді байланыс арнасы арқылы беру кезінде пайда болатын қателерді автоматты түрде анықтау және кейбір жағдайларда түзету. Хэмминг коды қарапайымдылығы мен тиімділігі арқасында қазіргі таңда да көптеген цифрлық байланыс және сақтау жүйелерінде қолданылады.

Хэмминг кодтарының басты идеясы - бастапқы ақпараттық биттерге қосымша бақылау биттерін (артық биттер) енгізу. Бұл биттер хабардың белгілі бір бөліктерін бақылап, қате пайда болған жағдайда оның нақты орнын анықтауға мүмкіндік береді. Мысалы, егер бір бит бұрмаланса, кодтың ішкі құрылымы арқылы дәл сол биттің индексі анықталады және ол автоматты түрде түзетіледі.

Хэмминг кодын құру үшін хабардағы ақпараттық биттерге қосымша бақылау биттері енгізіледі. Егер бастапқы хабарда kkk ақпараттық бит болса, онда бақылау биттерінің саны rrr келесі шартты қанағаттандыруы тиіс:

$2r \geq k + r + 1 2^r \geq k + r + 12r≥k+r+1$

Бұл формула бір қате туындаған жағдайда оның орнын анықтау үшін жеткілікті бақылау биттерінің санын анықтайды. Мысалы, егер 4 ақпараттық бит болса ($k=4k=4k=4$), онда 3 бақылау бит жеткілікті ($r=3r=3r=3$), себебі $2^3 = 8 \geq 4 + 3 + 1 = 823=8≥4+3+1=8$.

Кодтау процесінде бақылау биттері хабардағы белгілі бір позицияларға орналастырылады. Олар, әдетте, екілік жүйедегі 1, 2, 4, 8 және т.б. дәрежелерге сәйкес индекстерде орналасады. Әр бақылау бит хабардың белгілі бір позицияларын тексереді және олардың мәніне сәйкес логикалық паритет (таңдамалы қосынды) есептейді. Осылайша, барлық хабар биттері бақылау биттері арқылы өзара байланысады.

Хабар байланыс арнасы арқылы жіберілгеннен кейін қабылдаушы құрылғы бақылау биттерін қайта есептейді және алынған мәндерді салыстырады. Егер барлық бақылау биттері дұрыс болса, хабар бұзылмай жеткізілген деп есептеледі. Ал егер сәйкессіздік болса, онда ол қателік синдромы түрінде көрінеді - бұл синдром хабардағы қай бит бұзылғанын көрсететін екілік сан.

Хэмминг кодының үлкен артықшылығы - ол бір биттік қатені толықтай түзете алады және екі биттік қатені анықтай алады. Бұл қасиет оны сенімді

байланыс жүйелерінде, жад құрылғыларында және микроконтроллерлерде жиі қолдануға мүмкіндік береді. Мысалы, көптеген DRAM және Flash жадтарда Хэмминг коды қателерді түзету (ECC - Error Correcting Code) механизмдерінің негізін құрайды.

Сонымен қатар, Хэмминг кодының кеңейтілген нұсқасы да бар - кеңейтілген Хэмминг коды (extended Hamming code). Бұл кодқа қосымша бір жалпы паритеттік бит қосылады, ол екі және одан да көп биттік қателерді анықтауға мүмкіндік береді. Бұл тәсіл байланыс жүйесінің сенімділігін одан әрі арттырады.

Хэмминг кодтарының тиімділігі оның қарапайымдылығы мен есептеу тұрғысынан жеңілдігінде. Оны аппараттық немесе бағдарламалық деңгейде оңай жүзеге асыруға болады, себебі барлық операциялар тек логикалық қосу (XOR) негізінде орындалады. Бұл әдіс микроконтроллерлер мен кірістірілген жүйелерде энергияны аз тұтынатын шешімдердің бірі болып саналады.

Қорыта айтқанда, Хэмминг коды — ақпаратты бұрмалануданқорғаудың тиімді және математикалық тұрғыдан дәлелденген әдісі. Ол ақпараттың тұтастығын сақтауға, байланыс жүйелерінің сенімділігін арттыруға және қателердің әсерін азайтуға мүмкіндік береді. Осы себепті, ол қазіргі заманғы цифрлық коммуникацияның, компьютер жадының және деректерді сақтау жүйелерінің ажырамас бөлігіне айналған.

Хэмминг кодтарының негізгі идеялары, минималды қашықтықты есептеу және қателерді түзету принциптері

Хэмминг кодының негізгі идеясы - ақпараттық хабарды артық биттермен толықтырып, қателерді анықтау және түзетуге мүмкіндік беретін артықтық енгізу. Бұл артық биттер хабардың логикалық құрылымын сақтай отырып, оның тұтастығын бақылауға мүмкіндік береді. Кодтың басты мақсаты — хабардағы бір биттің бұрмалануын автоматты түрде тауып, оны түзету. Осы мақсатта кодтаудың және декодтаудың математикалық негізі ретінде екілік арифметика мен логикалық операциялар пайдаланылады.

Кодтау кезінде бастапқы хабар биттері белгілі бір позицияларға орналастырылады, ал бақылау биттері (parity bits) арнайы орындарға (2-нің дәрежесіне тең позицияларға) қойылады. Мысалы, 7 биттік Хэмминг кодында 4 ақпараттық және 3 бақылау бит болады: позициялар 1, 2 және 4 бақылау биттерге арналған, ал қалғандары — ақпараттық. Әр бақылау бит хабардың белгілі бір комбинациясын тексеріп, олардың логикалық қосындысы 0 немесе 1 болатындай етіп есептеледі.

Хэмминг кодтарының тиімділігі мен сенімділігі олардың минималды кодтық қашықтығына (dmin) байланысты. Минималды қашықтық — екі түрлі кодтық комбинация арасындағы ең аз айырмашылықтағы биттер саны. Бұл шама кодтың қателерге қарсы тұру қабілетін сипаттайды. Егер кодтың минималды қашықтығы dmind_{min}dmin болса, онда ол

- dmin−1d_{min} - 1dmin−1 қателерді анықтай алады,
- dmin−12\frac{d_{min} - 1}{2}2dmin−1 қателерді түзете алады.

Мысалы, қарапайым Хэмминг кодында dmin=3d_{min} = 3dmin=3, сондықтан ол бір биттік қатені түзете алады және екі биттік қатені анықтай

алады, бірақ оны түзете алмайды. Бұл қасиет Хэмминг кодын бірқатар сенімді байланыс жүйелерінде стандартты құрал ретінде пайдалануға мүмкіндік береді.

Хэмминг кодтарының артық биттері арнайы формула арқылы есептеледі:

$k + r + 1,2r \geq k+r+1$,

мұндағы rrr — бақылау биттерінің саны, kkk — ақпараттық биттер саны. Бұл формула кодтың әрбір ықтимал қатесін бірегей түрде анықтау үшін жеткілікті бақылау биттерінің санын табуға мүмкіндік береді.

Қате пайда болған жағдайда қате синдромы (error syndrome) есептеледі. Бұл синдром - қабылданған хабар мен бақылау биттері арқылы алынған логикалық қосынды нәтижесі. Егер синдром нөлге тең болса, хабар бұзылмаған; ал егер нөлден өзгеше мән шықса, онда оның екілік мәні хабардағы қатенің нақты орнын көрсетеді. Мысалы, егер синдром мәні 101_2 болса, онда қатенің индексі 5 екені анықталады, яғни бесінші бит бұрмаланған.

Қатені түзету процесі өте қарапайым: синдром арқылы қате орналасқан позиция анықталып, сол биттің мәні ($0 \rightarrow 1$ немесе $1 \rightarrow 0$) өзгертіледі. Осылайша, бастапқы хабар қалпына келтіріледі. Мұндай тәсіл қосымша күрделі есептеулерді қажет етпейді және аппараттық түрде оңай жүзеге асырылады.

Минималды қашықтықтың жоғары болуы жүйенің сенімділігін арттырады, бірақ сонымен бірге артық биттер саны да көбейеді. Сондықтан, кодтау тиімділігін арттыру үшін Хэмминг кодтары мен кеңейтілген кодтар (extended Hamming codes) арасындағы теңгерім табу қажет. Кеңейтілген нұсқада жалпы паритеттік бит қосылады, ол екі биттік қателерді де анықтауға мүмкіндік береді.

Қазіргі цифрлық байланыс жүйелерінде Хэмминг кодтары көптеген практикалық қолдануларға ие. Мысалы, олар жедел жадтағы (RAM) қателерді түзету үшін, спутниктік байланыста, сымсыз сенсорлық желілерде және микроконтроллерлік жүйелерде қолданылады. Бұл кодтар қателерді түзету үшін қарапайым, бірақ сенімді әдіс ұсынады, сондықтан олар энергия тиімділігі жоғары және ресурстары шектеулі құрылғыларда жиі пайдаланылады.

Қорыта айтқанда, Хэмминг кодының негізгі идеясы — ақпараттық хабарды бақылау биттері арқылы қорғау, ал оның математикалық негізі — минималды қашықтық және қате синдромы ұғымдары. Бұл код ақпаратты сенімді және тиімді түрде беруді қамтамасыз етеді, байланыс жүйелерінің тұрақтылығын арттырады және ақпараттық технологиялардағы маңызды қателікке қарсы әдіс болып табылады.

Хэмминг кодын практикалық тұрғыдан іске асыру кезінде басты мақсат — хабарды кодтау, арна арқылы жіберу және қабылданған хабарды қателерге тексеру мен қажет болған жағдайда түзету. Бұл әдіс көптеген микроконтроллерлерде, байланыс жүйелерінде және жады құрылғыларында қолданылады. Төменде Хэмминг кодын іске асырудың қарапайым мысалдары келтірілген.

Бірінші мысал ретінде (7,4) Хэмминг коды қарастырылады. Мұнда 4 ақпараттық бит және 3 бақылау бит қолданылады. Бақылау биттері хабардың келесі позицияларында орналасады: 1, 2 және 4. Егер бастапқы хабар 4 биттен тұрса, мысалы, M=[1,0,1,1]M = [1, 0, 1, 1]M=[1,0,1,1], онда бақылау биттерін есептеу қажет.

Бақылау биттері төмендегідей есептеледі:

- P1P_1P1 (1-орында) — 1, 3, 5, 7 позициялардағы биттерді тексереді;
- P2P_2P2 (2-орында) — 2, 3, 6, 7 позициялардағы биттерді тексереді;
- P4P_4P4 (4-орында) — 4, 5, 6, 7 позициялардағы биттерді тексереді.

Бақылау биттерінің мәндері логикалық XOR (қосу модулі 2) операциясы арқылы есептеледі:

P1=M1⊕M2⊕M4P_1 = M_1 \oplus M_2 \oplus M_4 P1=M1⊕M2⊕M4 P2=M1⊕M3⊕M4P_2 = M_1 \oplus M_3 \oplus M_4 P2=M1⊕M3⊕M4 P4=M2⊕M3⊕M4P_4 = M_2 \oplus M_3 \oplus M_4P4=M2⊕M3⊕M4

Нәтижесінде толық кодтық сөз, мысалы, былай көрінуі мүмкін:

C=[P1,P2,M1,P4,M2,M3,M4]=[0,1,1,0,0,1,1]C = [P_1, P_2, M_1, P_4, M_2, M_3, M_4] = [0, 1, 1, 0, 0, 1, 1]C=[P1,P2,M1,P4,M2,M3,M4]=[0,1,1,0,0,1,1]

Бұл хабар байланыс арнасы арқылы жіберіледі.

Егер арнада бір бит бұрмаланса, мысалы, 5-орындағы бит 0 → 1 болып өзгерсе, қабылданған код R=[0,1,1,0,1,1,1]R = [0, 1, 1, 0, 1, 1, 1]R=[0,1,1,0,1,1,1] болады. Енді қабылдаушы бақылау биттерін қайта есептеп, қате синдромын анықтайды. Қате синдромы үш биттен тұрады және қай позицияда қате болғанын көрсетеді. Егер синдром нәтижесі 101_2 болса, онда бұл 5-орындағы бит бұзылғанын білдіреді. Қабылдаушы бұл битті қайта ауыстырып, бастапқы хабарды қалпына келтіреді.

Бағдарламалық тұрғыдан бұл алгоритмді микроконтроллерде немесе кез келген бағдарламалау тілінде оңай жүзеге асыруға болады. Мысалы, C тіліндегі қысқаша псевдокод нұсқасы:

```
int encodeHamming(int d1, int d2, int d3, int d4) {
   int p1 = d1 ^ d2 ^ d4;
   int p2 = d1 ^ d3 ^ d4;
   int p4 = d2 ^ d3 ^ d4;
   int code = (p1 << 6) | (p2 << 5) | (d1 << 4) | (p4 << 3) | (d2 << 2) | (d3 << 1) | d4;
   return code;
}
```

Бұл код 4 ақпараттық биттен 7 биттік Хэмминг кодты қалыптастырады. Ал декодтау кезінде дәл осылай бақылау биттері қайта есептеліп, синдром арқылы қатенің орны анықталады.

Хэмминг кодтарын іске асырудың тағы бір маңызды саласы — жады құрылғыларындағы қателерді түзету (ECC — Error Correcting Code). DRAM, SRAM немесе Flash жадтарда ақпарат сақтау кезінде биттік қателердің пайда

болу ықтималдығы жоғары. Хэмминг коды мұндай жағдайда аппараттық деңгейде іске асырылады және әрбір жад ұяшығында артық биттер сақталып, оқылған деректердің дұрыстығы тексеріледі.

Сонымен қатар, бұл кодтау әдісі спутниктік байланыс, радио байланыс, аэроғарыш жүйелері сияқты сыртқы кедергілер жиі болатын ортада да кеңінен қолданылады. Мұндай жағдайларда хабардың сенімді берілуі аса маңызды, ал Хэмминг коды бір биттік қатені түзете алатындықтан, ол тиімді әрі есептеу тұрғысынан үнемді шешім болып табылады.

Қорытындылай келе, Хэмминг кодын іске асыру мысалдары оның әмбебаптығын және тиімділігін көрсетеді. Бұл әдіс аз ресурстарды талап ете отырып, жоғары сенімділікті қамтамасыз етеді. Сондықтан ол байланыс, есептеу және сақтау жүйелерінде ақпараттың тұтастығын қамтамасыз етудің негізгі құралдарының бірі болып табылады.

Пайдаланылған әдебиеттер тізімі

1. Шеннон, К. Э. *Математикалық коммуникация теориясы.* — М.: Наука, 1963.
2. Хаффман, Д. А. *A Method for the Construction of Minimum-Redundancy Codes.* — Proceedings of the IRE, 1952.
3. Фано, Р. М. *The Transmission of Information: A Statistical Theory of Communications.* — Cambridge, MIT Press, 1961.
4. Коберн, Д., Трембле, Г. *Теория кодирования и ее приложения.* — М.: Мир, 1982.
5. Mayo Венбо. *Прикладная криптография. Протоколы, алгоритмы и исходные тексты на языке С.* — М.: Триумф, 2004.