საქართველოს ტექნიკური
უნივერსიტეტი

ს. შავგულიძე, ნ. უღრელიძე,
თ. მთვრალაშვილი

ხელშეშლამდგრადი კოდირების თეორიული საფუძვლები

(სახელმძღვანელო)

თბილისი
2021

უაკ 621.391

ს. შავგულიძე, ნ. უღრელიძე, თ. მთვრალაშვილი. ხელშეშლამდგრადი კოდირების თეორიული საფუძვლები. სახელმძღვანელო. სტუ-ს გამომცემლობა. თბილისი, 2021 წ. 203 გვ.

სახელმძღვანელოში მოყვანილია ძირითადი ცნებები ბლოკური და უწყვეტი (ხისმაგვარი, გისოსისებრი, ხვევადი) კოდირების თეორიიდან, მათი აგების პრინციპები, ზოგიერთი ალბათური და დისტანციური მახასიათებლები, დეკოდირების ცნობილი ალგორითმები და ასეთი კოდების კონკრეტული მაგალითები. ნაშრომის პირველი ნაწილი (პარაგრაფები 2-11) განიხილავს ბლოკურ კოდებს, სადაც განსაკუთრებული ყურადღება გამახვილებულია რიდ-სოლომონის კოდებზე, ხოლო ნაშრომის მეორე ნაწილი (პარაგრაფები 12-25) განიხილავს უწყვეტ კოდებს.

წიგნი ძირითადად განკუთვნილია ტელეკომუნიკაციის სპეციალობის სტუდენტებისათვის. იგი აგრეთვე შეიძლება გამოიყენონ იმ სპეციალობის სტუდენტებმა, მაგისტრანტებმა და დოქტორანტებმა, რომელთაც საქმე აქვთ ინფორმაციის გადაცემას, მიღებასა და დამუშავებასთან.

ილუსტრაცია 23, ცხრილი 8, ლიტერატურა 34 დასახელება.

1. შესავალი

კავშირგაბმულობის სისტემათა უმრავლესობისათვის ბოლო ხანებში ჩვეულებრივ მოვლენად იქცა ხელშეშლამდგრადი კოდირების გამოყენება. ამასთან, კოდირება გამოიყენება არა მარტო კავშირის არხის გადაცემის სიზუსტის ამაღლებისა და ენერგეტიკული ეფექტურობის გაზრდისთვის, არამედ მისი საშუალებით შესაძლებელი ხდება ინფორმაციის გადაცემასთან დაკავშირებული მრავალი ამოცანის იდეურად ახლებურად გადაჭრა. მაგალითად, კოდური მეთოდების გამოყენებით შესაძლებელი გახდა სპექტრულად ეფექტური მაღალსიჩქარიანი სიგნალურ-კოდური კონსტრუქციების აგება, სიმბოლოთაშორისი ინტერფერენციის თავიდან აცილება, უწყვეტი ფაზის მქონე სიხშირულად მოდულირებული სიგნალების დემოდულაციის ხარისხის გაუმჯობესება, სიხშირეებზე მხტომი ახალი ტიპის სისტემის შემუშავება არხებში მრავალჯერადი დაშვებით და ა. შ.

ხელშეშლამდგრადი კოდები იყოფა ორ დიდ ჯგუფად: ბლოკურ და უწყვეტ კოდებად; იხ., მაგალითად, ნაშრომები [1]-[6]. ბლოკური კოდების კოდური მიმდევრობები შედგება ცალკეული კოდური კომბინაციებისაგან (ბლოკებისაგან), რომელთა კოდირება და დეკოდირება ხდება ერთმანეთისაგან დამოუკიდებლად. უწყვეტი კოდები და მათი სპეციალური კლასები - ხისმაგვარი, გისოსისებრი და ხვევადი კოდები წარმოადგენენ გამოკვლევისათვის გაცილებით უფრო ზოგად და, იმავდროულად, საინტერესო ობიექტს, ვიდრე ბლოკური კოდები. მართალია, კოდური მიმდევრობა ამ შემთხვევაშიც შედგება ელემენტარული ბლოკებისაგან, მათი

კოდირებისა და დეკოდირების პროცესებს აქვთ უწყვეტი ხასიათი და თითოეული კოდური ბლოკი დამოკიდებულია ერთ ან რამდენიმე წინა ასეთსავე ბლოკზე. როგორც გამოკვლევები გვიჩვენებენ, უწყვეტ კოდებს, კერძოდ, მათ პრაქტიკაში ყველაზე უფრო გავრცელებულ კლასს - ხვევად კოდებს აქვთ გაცილებით უკეთესი დისტანციური, ალბათური და ენერგეტიკული მახასიათებლები, ვიდრე ბლოკურ კოდებს ერთნაირი რეალიზაციის სირთულისას. ისინი უკეთ უთანხმდებიან არსებულ სიგნალთა სისტემებს და შემუშავებულია მათი კოდირებისა და დეკოდირების ალგებრული თუ სტატისტიკური ალგორითმები. ამჟამად ხვევადი კოდები ინტენსიურად შეისწავლება მსოფლიო ლიტერატურაში და ფართოდ გამოიყენება კავშირგაბმულობის სხვადასხვა სისტემაში, საიდანაც ისინი თანდათანობით დევნიან ბლოკურ კოდებს. ერთ-ერთ გამონაკლისს შეადგენს არაორობითი ბლოკური რიდ-სოლომონის კოდების კლასი, რომლებიც დღესაც წარმატებით გამოიყენება პრაქტიკაში, განსაკუთრებით კასკადური კოდირების სქემებში.

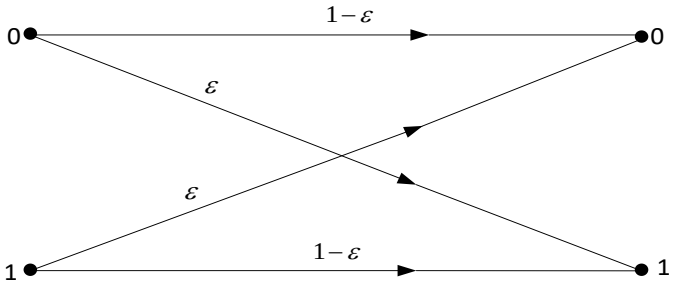
ჯერჯერობით ქართულ ენაზე არსებობს მხოლოდ რამდენიმე ნაშრომი (მათ შორის [7]-[9]), რომლებშიც მოცემულია კოდირების თეორიის ფუნდამენტური საკითხები, მაგრამ, სამწუხაროდ, არც ერთი მათგანი არ განიხილავს ერთდროულად, როგორც ბლოკურ, ისე უწყვეტ კოდებს. წინამდებარე სახელმძღვანელოში მოყვანილია ძირითადი ცნებები ბლოკური და უწყვეტი (ხისმაგვარი, გისოსისებრი, ხვევადი) კოდირების თეორიიდან, მათი აგების პრინციპები, ზოგიერთი ალბათური და დისტანციური მახასიათებლები, დეკოდირების ცნობილი ალგორითმები და ასეთი კოდების

კონკრეტული მაგალითები. ნაშრომის პირველი ნაწილი (პარაგრაფები 2-11) განიხილავს ბლოკურ კოდებს, სადაც განსაკუთრებული ყურადღება გამახვილებულია რიდ-სოლომონის კოდებზე, ხოლო ნაშრომის მეორე ნაწილი (პარაგრაფები 12-25) განიხილავს უწყვეტ კოდებს და წარმოადგენს ჩვენ მიერ ადრე გამოცემული დამხმარე სახელმძღვანელოს [9] გადამუშავებულ ვერსიას. მასალის ათვისებისათვის მკითხველს არ მოეთხოვება რაიმე სპეციალური ცოდნა კოდირების თეორიაში, თუმცა სასურველია იგი იცნობდეს ძირითად განმარტებებს [7]-[9]-დან.

2. ორი კოდური სიტყვის ექსპონენტა დისკრეტული უმეხსიერებო არხისათვის

არხს, სასრული შესასვლელი ალფაბეტით და სასრული გამოსასვლელი ალფაბეტით, რომელიც მოქმედებს ერთმანეთისაგან დამოუკიდებლად თითოეულ შესასვლელ სიმბოლოზე და რომლის სტატისტიკაც არ იცვლება დროის მიხედვით, ეწოდება დისკრეტული უმეხსიერებო არხი. დავუშვათ, რომ $A = \{a_1, a_2, \ldots, a_q\}$ და $B = \{b_1, b_2, ..., b_{q1}\}$ შესაბამისად შესასვლელი და გამოსასვლელი ალფაბეტებია. დისკრეტული უმეხსიერებო არხი მთლიანად განისაზღვრება $P(b_j \mid a_i)$ პირობითი ალბათობებით იმისა, რომ a_i -ის გადაცემისას მიღებულია b_j, სადაც $i = 1, 2, ..., q$ და $j = 1, 2, ..., q1.$ ასეთი არხი შეიძლება წარმოდგენილ იქნას მიმართული გრაფის საშუალებით, რომელშიც a_i და b_j კვანძების შემაერთე-

ბელი შტო ზემოდან აღნიშნულია $P(b_j \mid a_i)$ სიდიდით. პირველ ნახაზზე მოცემულია მიმართული გრაფი ორობითი სიმეტრიული არხისათვის, რომლისთვისაც $A = B = \{0,1\}$. ε სიდიდეს ეწოდება ამ არხის გადასასვლელი ალბათობა.



ნახ. 1. ორობითი სიმეტრიული არხი

(N, R) ბლოკური კოდი დისკრეტული უმეხსიერებო არხისათვის წარმოადგენს $m = 2^{NR}$ სიტყვისაგან შემდგარ მოწესრიგებულ სიმრავლეს, სადაც თითოეული სიტყვა შეიცავს A ალფაბეტის N სიმბოლოს. ამ სიტყვათა (ვექტორთა) სიმრავლე შეიძლება ჩაიწეროს შემდეგი სახით: $(\underline{x}_1, \underline{x}_2, ..., \underline{x}_m)$, სადაც $\underline{x}_j = \left[x_{j1}, x_{j2}, ..., x_{jN} \right]$. N და R პარამეტრებს ეწოდებათ, შესაბამისად, კოდის სიგრძე და სიჩქარე. R სიდიდე იზომება „გადაცემულ ბიტებში არხის თითოეული გამოყენებისას“, ვინაიდან როდესაც ყველა m კოდური სიტყვა თანაბარალბათურია, დისკრეტული უმეხსიერებო არხის N -ჯერ გამოყენებისას გადაიცემა $\log m = NR$ საინ-

ფორმაციო ბიტი (აქ და შემდგომში ყველგან იგულისხმება, რომ ლოგარითმი აღებულია 2-ის ფუძით).

აღვნიშნოთ კავშირის არხიდან მიღებული სიტყვა $\underline{y} = [y_1, y_2, ..., y_N]$ სიდიდით. მაქსიმალური დამაჯერებლობის დეკოდერი ირჩევს ისეთ კოდურ $\underline{x}_j (j = 1, 2, ..., m)$ სიტყვას, რომელიც ახდენს შემდეგი გამოსახულების მაქსიმიზაციას:

$$P(\underline{y} \mid \underline{x}_j) = \sum_{n=1}^{N} P(y_n \mid x_{jn}).$$

განვიხილოთ უმარტივესი ტიპის $(N, R = 1/N)$ კოდები, რომლებიც შედგება მხოლოდ $m = 2$ კოდური სიტყვისაგან. არხის შესასვლელ ალფაბეტზე ალბათობათა განაწილება აღვნიშნოთ Q სიდიდით. თითოეული $(\underline{x}_1, \underline{x}_2)$ კოდისათვის შეიძლება განისაზღვროს ალბათობა:

$$P(\underline{x}_1, \underline{x}_2) = \sum_{n=1}^{N} Q(x_{1n}) Q(x_{2n}),$$

რომელიც წარმოადგენს კოდის არჩევის ალბათობას თუ თითოეული სიმბოლო კოდურ სიტყვებში აირჩევა ერთმანეთისაგან დამოუკიდებლად Q-ს შესაბამისად. $(\underline{x}_1, \underline{x}_2)$ კოდის მაქსიმალური დამაჯერებლობით დეკოდირებისას შეცდომის ალბათობა აღვნიშნოთ $P_e(\underline{x}_1, \underline{x}_2)$ სიდიდით. მაშინ გამოსახულება:

$$\overline{P_e} = \sum_{\underline{x}_1 \in A^N} \sum_{\underline{x}_2 \in A^N} P_e(\underline{x}_1, \underline{x}_2) P(\underline{x}_1, \underline{x}_2)$$

განსაზღვრავს მაქსიმალური დამაჯერებლობით დეკოდირებისას შეცდომის საშუალო ალბათობას კოდების ისეთი ანსამ-

ბლისათვის, რომელშიც თითოეული კოდი შეიცავს N სიგრძის მქონე $m = 2$ სიტყვას.

ადვილად შეიძლება ვუჩვენოთ [5], რომ იმის მიუხედავად, თუ რა წესით ვირჩევთ სიტყვებს მოცემულ ორსიტყვიან კოდში, გვაქვს:

$$\overline{P_e} \le 2^{-NR_0}, \tag{2.1}$$

სადაც

$$R_0 = -\log\{\min_Q \sum_{y\in B}[\sum_{x\in A} Q(x)\sqrt{p(y\mid x)}]^2\}. \tag{2.2}$$

ამრიგად, Q არჩეულ უნდა იქნას ისე, რომ მოახდინოს (2.2) ფორმულით მოცემულ ალბათობათა განაწილების მინიმიზაცია. ვინაიდან უნდა არსებობდეს ერთი კოდი მაინც, რომლისთვისაც $P_e(\underline{x}_1, \underline{x}_2)$ სიდიდე იქნება არა უარესი, ვიდრე შეცდომის საშუალო ალბათობა კოდების მთელი ანსამბლისათვის, შესაბამისად (2.1) ფორმულიდან გამომდინარე არსებობს ისეთი კოდების მიმდევრობა, რომელთათვისაც მაქსიმალური დამაჯერებლობით დეკოდირებისას შეცდომის ალბათობა არ აღემატება 2^{-NR_0} სიდიდეს. ამრიგად, ეს ალბათობა თავის მხრივ ექსპონენციალურად მცირდება N-ის ზრდასთან ერთად. ექსპონენტის R_0 მაჩვენებელს, რომელიც განსაზღვრულია (2.2) გამოსახულებით, ეწოდება ორი კოდური სიტყვის ექსპონენტა დისკრეტული უმეხსიერებო არხისთვის.

ორობითი $A = \{0,1\}$ უმეხსიერებო არხისათვის R_0-ის გამოთვლა მარტივდება იმის გამო, რომ $Q(0) = Q(1) = 1/2$ ყოველთვის არის განაწილება, რომელიც ახდენს (2.2) ფორმუ-

ლის მინიმიზაციას. მაგალითად, ორობითი სიმეტრიული არხისათვის

$$R_0 = 1 - \log[1 + 2\sqrt{\varepsilon(1-\varepsilon)}]. \qquad (2.3)$$

(2.3)-დან გამომდინარე, როდესაც, მაგალითად, ε უდრის 0.57, 0.45 და 0.33, R_0 შესაბამისად ტოლია 0.45, 0.50 და 0.55.

მომდევნო პარაგრაფებში R_0 სიდიდე გამოყენებული იქნება მაქსიმალური დამაჯერებლობით დეკოდირებისას $\overline{P_e}$ სიდიდის საზღვრების საპოვნელად გაცილებით უფრო საინტერესო კოდების ანსამბლებისათვის, ვიდრე ორსიტყვიანი კოდებისათვის. ამიტომ თავდაპირველად ჩვენ უფრო სიღრმისეულად განვიხილავთ ბლოკურ კოდებს და მათი კოდირება-დეკოდირების ალგორითმებს.

3. ბლოკური კოდირება

ბლოკური კოდირება-დეკოდირების ძირითადი პრინციპების შესასწავლად ჩვენ ვისარგებლებთ შემდეგი განსაზღვრებით.

განსაზღვრება 3.1. მოდულით ოპერაცია $a = cb + d$ სიდიდისათვის განისაზღვრება როგორც $a = d \bmod b$, სადაც $a, c \in Z, b \in N$ ხოლო $d \in N_0$ (გამოითქმება ასე: a d სიდიდის კონგრუენტულია b -ს მოდულით).

ზემოთ მოყვანილ განსაზღვრებაში Z, N, და N_0 შესაბამისად აღნიშნავენ მთელი, ნატურალური და მთელი არაუარყოფითი რიცხვების სიმრავლეს.

მაგალითი 3.1. განვიხილოთ მოდულით ოპერაციის რამდენიმე მაგალითი:

$$71 = 1 \bmod 7,\ \ -13 = 2 \bmod 5,\ \ 30 = 6 \bmod 8,$$

$$(25 \cdot 31) = 4 \cdot 3 = 5 \bmod 7.$$

შემოვიტანოთ ბლოკური კოდის კიდევ ერთი, კლასიკური განმარტება: C ბლოკური კოდი k სიგრძის მქონე საინფორმაციო $i_0, i_1, ..., i_{k-1}$ სიმბოლოთა ბლოკს ცალსახად ასახავს n სიგრძის მქონე კოდურ $c_0, c_1, ..., c_{n-1}$ სიტყვაში. ასახვის დროს გამოყენებულ ჭარბ სიმბოლოთა რიცხვი ტოლია $(n-k)$-სი. k/n ფარდობა განსაზღვრავს კოდის სიჩქარეს. თუ ორობით საინფორმაციო ბიტთა მიმდევრობა კოდირებისას იყოფა დამოუკიდებელ ბლოკებად, მაშინ საქმე გვაქვს ორობით ბლოკურ კოდთან, სადაც 0 და 1 სიმბოლოები შეადგენენ კოდის ალფაბეტს.

მაგალითი 3.2. განვიხილოთ უმარტივესი ლუწობის შემმოწმებელი ბლოკური კოდი. საინფორმაციო $i_0, i_1, ..., i_{k-1}$ ბლოკი, რომელიც შეიცავს k ორობით სიმბოლოს, აისახება $n = k+1$ სიგრძის $c_0, c_1, ..., c_{n-1}$ კოდურ სიტყვაში; ამასთან $c_0 = i_0, c_1 = i_1, ..., c_{k-1} = i_{k-1}$, ხოლო კოდური სიტყვის ბოლო c_{n-1} სიმბოლო აკმაყოფილებს $\sum_{j=0}^{n-2} i_j + c_{n-1} = 0 \bmod 2$ ტოლობას. ქვემოთ მოყვანილი ცხრილი 1 წარმოგვიდგენს კოდურ სიტყვებს $n = 3$ შემთხვევისათვის.

მაგალითი 3.3. n სიგრძის მქონე კოდი განმეორებით შეიცავს ორ კოდურ სიტყვას: პირველი მათგანი შედგება მხო-

ლოდ ნულიანებისგან $c_0 = c_1 = ... = c_{n-1} = 0,$ ხოლო მეორე მათგანი შედგება ერთიანებისგან $c_0 = c_1 = ... = c_{n-1} = 1.$

ცხრილი 1

i_0	i_1	c_2
0	0	0
0	1	1
1	0	1
1	1	0

განსაზღვრება 3.2. ორი კოდური სიტყვის შეკრების ოპერაცია $\underline{c} + \underline{a},$ განისაზღვრება ამ სიტყვების j -ური კოორდინატების შეკრების $c_j + a_j,$ $j = 0,1,...,n-1,$ მეშვეობით, სადაც თითოეული კოორდინატისათვის მიღებულ ჯამზე სრულდება ოპერაცია 2-ის მოდულით.

განსაზღვრება 3.3. კოდს ეწოდება წრფივი, თუ კოდური სიტყვების ყველა წრფივი კომბინაცია (მაგალითად, მათი შეკრება) ისევ გვაძლევს კოდურ სიტყვას.

ლუწობის შემმოწმებელი ბლოკური კოდი (მაგალითი 3.2) და კოდი განმეორებით (მაგალითი 3.3) წარმოადგენენ წრფივ კოდებს.

ორობითი ბიტების მიმდევრობა შეიძლება წარმოდგენილ იქნას როგორც ორობითი კომპონენტების შემცველი

ერთ-ერთი ვექტორი გალუას ველის (GF) F_2^n ვექტორული სივრცის შემადგენელი n სიგრძის მქონე ყველა შესაძლო ვექტორთა ერთობლიობიდან [1].

განსაზღვრება 3.4. $\underline{a},\underline{b} \in F_2^n$ ვექტორების სკალარული ნამრავლი განისაზღვრება შემდეგნაირად:

$$\langle \underline{a},\underline{b} \rangle = \sum_{i=0}^{n-1} a_i b_i \bmod 2.$$

განსაზღვრება 3.5. $\underline{c}$ ვექტორის ჰემინგის წონა (wt) განისაზღვრება როგორც ამ ვექტორის არანულოვან კოორდინატთა რიცხვი:

$$wt(\underline{c}) = \sum_{j=0}^{n-1} wt(c_j), \text{ სადაც } wt(c_j) = \begin{cases} 0, & c_j = 0, \\ 1, & c_j \neq 0. \end{cases}$$

განსაზღვრება 3.6. ორ $\underline{a}$ და $\underline{c}$ ვექტორს შორის ჰემინგის მანძილი ($dist$) ტოლია კოორდინატთა რიცხვის, რომელშიც $\underline{a}$ და $\underline{c}$ განსხვავდება ერთმანეთისაგან:

$$dist(\underline{a},\underline{c}) = \sum_{j=0}^{n-1} wt(a_j + c_j), \text{ სადაც } wt(a_j + c_j) = \begin{cases} 0, & c_j = a_j, \\ 1, & c_j \neq a_j. \end{cases}$$

$$dist(\underline{a},\underline{c}) = wt(\underline{a} + \underline{c}).$$

განსაზღვრება 3.7. n სიგრძის მქონე C კოდისათვის წონათა აღნუსხვის $W = (w_0, w_1, ..., w_n)$ ვექტორის კოორდინატები განისაზღვრება როგორც j წონის მქონე სიტყვათა w_j რიცხვი. ეს ვექტორი შეიძლება ჩაწერილ იქნას შემდეგი მრავალწევრის სახით:

$$W(x) = \sum_{j=0}^{n} w_j x^j.$$

წრფივი C კოდი აუცილებლად შეიცავს ნულებისაგან შემდგარ კოდურ სიტყვას, $w_0 = 1$. ნულოვანი კოდური სიტყვის არსებობა აუცილებელია კოდის წრფივობის გამო $(\underline{c} + \underline{c} = \underline{0} \in C)$.

<u>მაგალითი 3.4.</u> განვიხილოთ 3.2 მაგალითში მოცემული $n = 3$ სიგრძის მქონე ლუწობის შემმოწმებელი კოდი. მისთვის წონათა აღნუსხვის ვექტორი ტოლია $W = (1, 0, 3, 0)$.

3.3 მაგალითში მოცემული კოდისათვის განმეორებით გვაქვს: $w_0 = 1;\ \ w_n = 1;\ \ w_j = 0,\quad j = 1, 2, ..., n-1.$

განსაზღვრება 3.8. C კოდისათვის მინიმალური d მანძილი განისაზღვრება როგორც ორ განსხვავებულ კოდურ სიტყვას შორის მინიმალური მანძილი:

$$d = \min_{\substack{\underline{a}, \underline{c} \in C \\ \underline{a} \neq \underline{c}}} \{dist(\underline{a}, \underline{c})\}.$$

წრფივი კოდებისათვის მინიმალური მანძილი ტოლია მინიმალური არანულოვანი წონისა:

$$d = \min_{\substack{\underline{a}, \underline{c} \in C \\ \underline{a} \neq \underline{c}}} \{wt(\underline{a} + \underline{c})\} = \min_{\substack{\underline{c} \in C \\ \underline{c} \neq 0}} \{wt(\underline{c})\}.$$

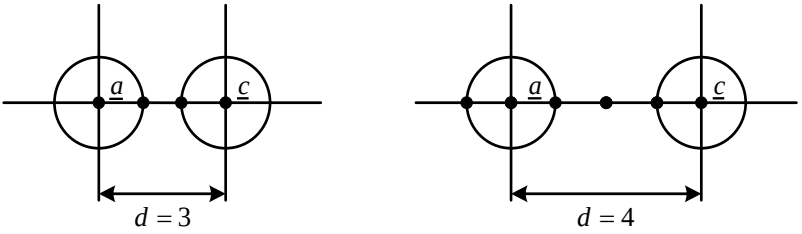
ის თვისება, რომ მინიმალური მანძილი ტოლია მინიმალური არანულოვანი წონის ძალზე მნიშვნელოვანია წრფივი კოდების აგებისას. კოდურ სიტყვებს შორის მინიმალური

მანძილი განსაზღვრავს კოდის შეცდომების გამამჟღავნებელ და გამასწორებელ შესაძლებლობებს. ამასთან, მინიმალური მანძილის გამოთვლა დაიყვანება მინიმალური არანულოვანი წონის გამოთვლაზე, რაც მნიშვნელოვნად ამარტივებს ამ პროცედურას.

ბუნებრივია, იბადება კითხვა: რა რაოდენობის შეცდომები შეიძლება გაასწოროს C კოდმა, რომლის მინიმალური მანძილია d ? ჩვენ განვიხილავთ ორ სიტყვას, რომლებიც C კოდში იმყოფება d -ს ტოლ მინიმალურ მანძილზე. როდესაც გადაცემულ კოდურ სიტყვაში ადგილი აქვს ერთადერთ შეცდომას, მაშინ მიღებული სიტყვა იმყოფება ერთის ტოლ მანძილზე გადაცემული კოდური სიტყვიდან. მე-2 ნახაზზე მოყვანილია მაგალითები, რომლებიც გვიჩვენებენ კოდის მიერ შეცდომების გასწორების შესაძლებლობას. აქ ორ მეზობელ წერტილს შორის ჰემინგის მანძილი ტოლია ერთის.

მე-2 ნახაზის მარცხენა დიაგრამაზე მოცემულია ორი $\underline{a}$ და $\underline{c}$ კოდური სიტყვა, რომელთა შორის ჰემინგის მანძილი ტოლია 3-ის, რაც შეესაბამება C კოდის მინიმალურ მანძილს. ამგვარად, მინიმალური მანძილის განსაზღვრების მიხედვით C კოდის ნებისმიერ ორ სიტყვას შორის ჰემინგის მანძილი მეტი ან ტოლია 3-ის. თუ ჩვენ თითოეულ კოდურ სიტყვას შემოვსაზღვრავთ ჰემინგის მანძილის მიხედვით ერთეულოვანი რადიუსის მქონე სფეროებით, ადვილად დავრწმუნდებით რომ ეს სფეროები არ კვეთენ ერთმანეთს (აღვნიშნავთ, რომ ნახაზის შეზღუდული შესაძლებლობების გამო, სფეროები მოცემულია წრეწირების სახით). ამგვარად, ნებისმიერი ვექტორი რომელიც მოთავსებულია კოდური

სიტყვიდან ერთზე ნაკლებ ან ერთის ტოლ ჰემინგის მანძილზე ცალსახად აისახება ამ კოდურ სიტყვაში.



ნახ. 2. ჰემინგის მანძილები

მე-2 ნახაზის მარჯვენა დიაგრამაზე განხილული შემთხვევისათვის მინიმალური მანძილი ტოლია: $d = 4$. ამგვარად, თუ მიღებული ვექტორი იმყოფება კოდური სიტყვისაგან ორის ტოლ ჰემინგის მანძილზე, მაშინ ის შეიძლება თანაბრად იყოს დაშორებული ორი განსხვავებული კოდური სიტყვისაგან. ეს გარემოება შეუძლებელს ხდის მიღებული ვექტორის კოდურ სიტყვაში ცალსახად ასახვის პროცესს, ვინაიდან მიღებული ვექტორი, რომელიც მოთავსებულია ორის ტოლ ჰემინგის მანძილზე $\underline{a}$ და $\underline{c}$ კოდური სიტყვებიდან, თანაბარი შესაძლებლობით, შეიძლება იქნას ასახული ერთ-ერთ მათგანში.

ამგვარად, ადგილი აქვს მხოლოდ შეცდომების გამჟღავნებას და არა მათ გასწორებას. მაშასადამე, ამ შემთხვევაშიც ჩვენ შეგვიძლია გავასწოროთ მხოლოდ ერთეულოვანი შეცდომები, ანუ ნებისმიერი ვექტორი, რომელიც კოდური

სიტყვიდან მოთავსებულია ერთზე ნაკლებ ან ერთის ტოლ ჰემინგის მანძილზე, ცალსახად ავსახოთ ამ კოდურ სიტყვაში.

ჩვენ განვაზოგადებთ ამ კონცეფციას: მიღებული ვექტორი $\underline{r} = \underline{c} + \underline{f}$ ($\underline{c} \in C$, $\underline{f}$ - შეცდომის ვექტორია), შეიძლება გასწორებულ იქნას თუ მანძილი ამ ვექტორსა და ნებისმიერ სხვა $\underline{a}$ კოდურ სიტყვას ($\underline{a} \in C$) შორის აკმაყოფილებს პირობას:

$$dist(\underline{c}, \underline{c} + \underline{f}) < dist(\underline{a}, \underline{c} + \underline{f}) \quad \text{ან} \quad wt(\underline{f}) < wt(\underline{a} + \underline{c} + \underline{f}).$$

თუ ეს შედეგი მართებულია გარკვეული წონის მქონე შეცდომათა სრული სიმრავლისათვის $\{\underline{f}\}$ მაშინ მივიღებთ:

$$wt(\underline{f}) \le \left\lfloor \frac{d-1}{2} \right\rfloor,$$

სადაც $\lfloor x \rfloor$ აღნიშნავს უდიდეს მთელ რიცხვს, რომელიც ნაკლებია ან ტოლი x-ის. ამგვარად, როდესაც d კენტია, მაშინ $\left\lfloor \frac{d-1}{2} \right\rfloor = \frac{d-1}{2}$, ხოლო $\left\lfloor \frac{d-1}{2} \right\rfloor = \frac{d-2}{2}$ მაშინ, როდესაც d ლუწია.

ჩვენ ვითვალისწინებთ იმ ფაქტს, რომ როდესაც არხში შეცდომის ალბათობა $\varepsilon = p < 0.5$, შეცდომების მცირე რაოდენობა უფრო მეტად ალბათურია, ვიდრე მათი დიდი რაოდენობა. სწორედ აქედან გამომდინარეობს დეკოდირების სტრატეგია - მიღებული ვექტორის ასახვა ჰემინგის მანძილის მიხედვით უახლოეს კოდურ სიტყვაში.

მაგალითი 3.5. უმარტივეს ლუწობის შემმოწმებელ კოდს (მაგალითი 3.2) აქვს მინიმალური მანძილი $d = 2$ რაც გვაძლევს $\left\lfloor \frac{d-1}{2} \right\rfloor = 0.$ ამგვარად, ასეთი უმარტივესი კოდით შეცდომების გასწორება შეუძლებელია. მეორე მხრივ, მიღებულ ვექტორში შეცდომების კენტი რიცხვი ყოველთვის გამჟღავნდება, ვინაიდან ლუწობის შემმოწმებელი ჯამი განსხვავდება ნულისაგან.

მაგალითი 3.6. კოდს განმეორებით (მაგალითი 3.3) აქვს მინიმალური მანძილი $d = n,$ ამგვარად, $\left\lfloor \frac{d-1}{2} \right\rfloor = \frac{n-1}{2},$ როდესაც n კენტია და $\left\lfloor \frac{d-1}{2} \right\rfloor = \frac{n-2}{2},$ როდესაც n ლუწია.

ორობითი სიმეტრიული არხის შემთხვევაში, კოდისათვის განმეორებით შეიძლება შემუშავებულ იქნას დეკოდირების შემდეგი მეთოდი. დავითვალოთ ნულების რაოდენობა მიღებულ სიტყვაში. თუ ეს რიცხვი მეტია $\left\lfloor \frac{n-1}{2} \right\rfloor$ სიდიდეზე მოვახდინოთ 0-ის დეკოდირება, საწინააღმდეგო შემთხვევაში დეკოდირების შედეგად მივიჩნიოთ 1. ტოლობის შემთხვევაში (ე. ი. $\frac{n}{2}$ და n ლუწია) შეგვიძლია მივიღოთ ნებისმიერი გადაწყვეტილება ან უბრალოდ აღვნიშნოთ შეცდომების გამჟღავნების ფაქტი.

ამგვარად, როდესაც $\underline{c} + \underline{f}$ არ არის კოდური სიტყვა, მაშინ შეიძლება შეცდომების გამჟღავნება. მეორე მხრივ, ნების-

მიერი $\underline{f} \neq \underline{0}$ -სთვის, რომლის წონაც $wt(\underline{f}) \leq d$, $\underline{c} + \underline{f}$ არ შეიძლება იყოს კოდური სიტყვა.

ყოველივე ზემოთქმულიდან გამომდინარე, ნებისმიერ $C(n,k,d)$ წრფივ კოდს შეუძლია $e = \left\lfloor \frac{d-1}{2} \right\rfloor$-ის ტოლი ან ნაკლები რაოდენობის შეცდომების გასწორება და ასევე შეუძლია $(d-1)$-ის ტოლი ან ნაკლები რაოდენობის შეცდომების გამჟღავნება.

კოდირების თეორიის ერთ-ერთ ძირითად პრობლემას წარმოადგენს კოდის მოცემული n სიგრძისა და მინიმალური d მანძილისათვის კოდური სიტყვების რაოდენობის დადგენა. ბუნებრივია ჩნდება კითხვა: კოდის აღნიშნული პარამეტრებისათვის როგორ შეიძლება განისაზღვროს კოდური სიტყვების რაოდენობის ზედა და ქვედა საზღვარი? ჰემინგის საზღვარი წარმოადგენს ზედა საზღვარს. მოცემული $\underline{c} \in C(n,k,d)$ ვექტორისათვის არსებობს 1-ის ტოლ მანძილზე მყოფი $\binom{n}{1}$ ვექტორი, 2-ის ტოლ მანძილზე მყოფი $\binom{n}{2}$ ვექტორი და ა. შ. $\binom{n}{t}$ ბინომიალური კოეფიციენტები განისაზღვრება შემდეგნაირად:

$$\binom{n}{t} = \frac{n(n-1)\cdot\ldots\cdot(n-t+1)}{t(t-1)\cdot\ldots\cdot 1}.$$

სულ გვაქვს 2^n ორობითი ვექტორი, სადაც n კოდის სიგრძეა. ამგვარად, ადვილად ვრწმუნდებით შემდეგი

თეორემის სამართლიანობაში, რომელიც ცნობილია როგორც ჰემინგის საზღვარი.

თეორემა 3.1. ორობითი კოდისათვის ადგილი აქვს შემდეგ უტოლობას:

$$2^k\left(1+\binom{n}{1}+.....+\binom{n}{e}\right)\le 2^n,$$

სადაც $e=\left\lfloor \frac{d-1}{2} \right\rfloor$.

ჰემინგის საზღვარს აქვს შემდეგი ინტერპრეტაცია. ნებისმიერი კოდური სიტყვა შეიძლება შემოსაზღვრული იყოს შეცდომების გამასწორებელი სფეროთი, რომელსაც აქვს ისეთი მაქსიმალური შესაძლო დიამეტრი, რომ სფეროები არ კვეთდეს ერთმანეთს. აღნიშნული მაქსიმალური დიამეტრი შეესაბამება კოდურ სიტყვებს შორის მინიმალურ მანძილს (იხ. მე-2 ნახაზი). ყველა ვექტორი, რომელიც მოთავსებულია გამასწორებელი სფეროს შიგნით, ცალსახად აისახება სფეროს ცენტრში მოთავსებულ კოდურ სიტყვაში.

განსაზღვრება 3.9. კოდს, რომელიც აკმაყოფილებს ჰემინგის საზღვარს, ეწოდება სრულყოფილი კოდი.

კოდირების თეორიიდან ცნობილია, რომ არსებობს მხოლოდ რამდენიმე სრულყოფილი კოდი, კერძოდ, კოდი განმეორებით, რომლის სიგრძე კენტი რიცხვია, ასევე ჰემინგისა და გოლეის კოდები [1].

ჰემინგის საზღვრის მიხედვით შეიძლება გამასწორებელი სფეროს შიგნით მოთავსებულ ვექტორთა რიცხვისა და F_2^n სივრცეში არსებული ვექტორთა სრული რიცხვის შედარება. საზოგადოდ, ჰემინგის საზღვარი აფასებს, თუ რამდე-

ნად კარგად არის ვექტორული სივრცე შევსებული გამასწორებელი სფეროებით. სრულყოფილი კოდების შემთხვევაში, გამასწორებელი სფეროები სრულად მოიცავენ ვექტორულ სივრცეს. ამგვარად, ყველა ვექტორი მოთავსებულია გამასწორებელი სფეროს შიგნით და შეიძლება ცალსახად აისახოს კოდურ სიტყვაში. აქედან გამომდინარე ჰემინგის საზღვარს ასევე უწოდებენ მჭიდროდ შეფუთვის საზღვარს.

<u>მაგალითი 3.7.</u> ორობითი ლუწობის შემმოწმებელი კოდისათვის ჰემინგის საზღვარი გამოითვლება (მაგალითი 3.2) შემდეგნაირად: $k = n-1,\ e = 0$ და $2^{n-1}\cdot(1) < 2^n$; მაშასადამე, კოდი არ არის სრულყოფილი.

<u>მაგალითი 3.8.</u> $n = 3$ სიგრძის მქონე კოდისათვის განმეორებით (იხ. მაგალითი 3.3) გვაქვს $k = 1,\ e = 1,$ $2^1\cdot(1+\binom{3}{1}) = 2^1\cdot 4 = 8 = 2^3$; მაშასადამე, კოდი სრულყოფილია.

თეორემა 3.2. ყველა კოდი განმეორებით, რომლის სიგრძე კენტი რიცხვია წარმოადგენს სრულყოფილ კოდს.

დამტკიცება: ყველა ასეთ n სიგრძის მქონე კოდს აქვს პარამეტრები $C(n,1,\frac{n-1}{2})$. თუ გამოვიყენებთ $\sum_{j=0}^{n}\binom{n}{j} = 2^n$ და $\binom{n}{j} = \binom{n}{n-j}$ ტოლობებს ჩვენ შეგვიძლია გამოვთვალოთ ჰემინგის საზღვარი:

$$2\cdot(1+\binom{n}{1}+\ldots+\binom{n}{\frac{n-1}{2}})=1+\binom{n}{1}+\ldots+\binom{n}{\frac{n-1}{2}}+\ldots+\binom{n}{1}+1$$

$$=1+\binom{n}{1}+\ldots+\binom{n}{\frac{n-1}{2}}+\binom{n}{\frac{n+1}{2}}+\ldots+\binom{n}{n-1}+\binom{n}{n}=\sum_{j=0}^{n}\binom{n}{j}$$

$$=2^n \qquad \square$$

წრფივი ბლოკური კოდის სიტყვები $\underline{c}=(c_0,c_1,\ldots,c_{n-1})$ შეიძლება განვსაზღვროთ შემდეგი განტოლებით:

$$H\underline{c}^T=0 \text{ ან } \underline{c}H^T=0,$$

სადაც H ლუწობის შემმოწმებელი მატრიცაა, ხოლო T ტრანსპონირების ნიშანია. n სიგრძისა და k განზომილების (საინფორმაციო ბლოკის სიგრძის) მქონე ორობითი ბლოკური კოდის შემთხვევაში H წარმოადგენს $(n-k)\times n$ მატრიცას. H მატრიცის $\underline{c}^T$ ვექტორზე გადამრავლების შედეგად მიიღება ვექტორი, რომლის მნიშვნელობებიც შეესაბამება H -ის სტრიქონების სკალარულ ნამრავლს $\underline{c}^T$ -ზე. H მატრიცისა და d მინიმალური მანძილის მიხედვით კოდის განსაზღვრებიდან გამომდინარეობს, რომ H მატრიცის ნებისმიერი $d-1$ სვეტი უნდა იყოს წრფივად დამოუკიდებელი, ხოლო ზოგიერთი d სვეტი წრფივად დამოკიდებულია.

<u>მაგალითი 3.9.</u> ლუწობის შემმოწმებელი კოდისათვის, რომელიც განხილული იყო 3.2 მაგალითში, ლუწობის შემმოწმებელ მატრიცას აქვს სახე:

$$H=(1,1,\ldots,1)$$

და, ვინაიდან $n-k=1$, H წარმოადგენს $(1\times n)-$მატრიცას.

n სიგრძის მქონე კოდს განმეორებით აქვს ლუწობის შემმოწმებელი მატრიცა:

$$H = \begin{pmatrix} 1 & 1 & & \\ 1 & & 1 & \\ \vdots & & & \ddots \\ 1 & & & 1 \end{pmatrix},$$

სადაც თავისუფალი პოზიციები შევსებულია ნულებით. ამგვარად, H წარმოადგენს $(n-1)\times n$ მატრიცას.

ლუწობის შემმოწმებელი H მატრიცის ნებისმიერი სტრიქონისა და ნებისმიერი ტრანსპონირებული კოდური სიტყვის ნამრავლი უნდა იყოს ნულის ტოლი. ამასთან H მატრიცის სტრიქონების წრფივი კომბინაცია გვაძლევს ლუწობის შემმოწმებელ სხვა H' მატრიცას, რომელიც განსაზღვრავს იგივე კოდს, რასაც H მატრიცა.

განსაზღვრება 3.10. საინფორმაციო სიმბოლოების კოდურ სიტყვაში ასახვას ეწოდება სისტემატური თუ k საინფორმაციო სიმბოლო შეუცვლელი რჩება კოდური სიტყვის n კოორდინატთა შორის. აქედან გამომდინარე, კოდურ სიტყვაში საინფორმაციო სიმბოლოები და სიჭარბის სიმბოლოები განლაგებულია სხვადასხვა პოზიციაზე. ლუწობის შემმოწმებელ მატრიცას აქვს შემდეგი სახე:

$$H = (A \mid I),$$

სადაც I $(n-k)\times(n-k)$ ზომების მქონე ერთეულოვანი მატრიცაა.

ნებისმიერი წრფივი ბლოკური კოდის კოდირება შესაძლებელია სისტემატური ფორმითაც.

განსაზღვრება 3.11. ტრანსპონირებული $\underline{s}$ სინდრომი განისაზღვრება როგორც მიღებული $\underline{r} = \underline{c} + \underline{f}$ სიტყვის ტრანსპონირებული ვექტორის ნამრავლი ლუწობის შემმოწმებელ მატრიცაზე, სადაც $\underline{c} \in C$, ხოლო $\underline{f}$ შეცდომის ვექტორია:

$$\underline{s}^T = H\underline{r}^T = H(\underline{c}^T + \underline{f}^T) = H\underline{f}^T.$$

ამგვარად, სინდრომი დამოკიდებულია მხოლოდ შეცდომის ვექტორზე და არა კოდურ სიტყვაზე. ეს გამომდინარეობს იმ ფაქტიდან, რომ $H\underline{c}^T = 0$ ნებისმიერი $\underline{c}$ -სთვის, $\underline{c} \in C$.

დეკოდირება შეიძლება განხორციელდეს ორ ეტაპად. თავდაპირველად განისაზღვრება მიღებული სიტყვის სინდრომი, ხოლო შემდეგ სინდრომის ბაზაზე იძებნება ყველაზე უფრო მეტად ალბათური შეცდომის ვექტორი.

4. ბლოკური კოდების დეკოდირების პრინციპები

არხიდან მიღებული სიტყვის დეკოდირებისას ადგილი შეიძლება ჰქონდეს სამ შემთხვევას: დეკოდირებული სიტყვა სწორია, მცდარია ან გადაწყვეტილება არ არის მიღებული. განვიხილოთ ცალ-ცალკე თითოეული შემთხვევა: სწორად დეკოდირებისას გადაცემული კოდური სიტყვა ემთხვევა დეკოდირების შედეგად მიღებულ სიტყვას. ამგვარად, დეკოდერმა მოახერხა გადაცემული სიტყვიდან მოეცილებინა არხის მიერ გენერირებული შეცდომები. მცდარად დეკოდირებისას გადაცემული კოდური სიტყვა არ ემთხვევა დეკოდირების შედეგად მიღებულ სიტყვას. ამ შემთხვევაში დეკოდი-

რების მიერ გათვლილი შეცდომის ვექტორი არ ემთხვევა არხის მიერ რეალურად გენერირებულ შეცდომის ვექტორს. მაგალითად, როდესაც შეცდომის ვექტორი ტოლია ერთერთი კოდური სიტყვის, კოდის წრფივობის გამო მიღებული ვექტორი ასევე ტოლი იქნება კოდური სიტყვის. ზოგიერთი დეკოდირების მეთოდის გამოყენებისას დეკოდერი ვერ პოულობს სწორ კოდურ სიტყვას და ამჯობინებს საერთოდ არ მიიღოს გადაწყვეტილება შესაძლო გადაცემული კოდური სიტყვის შესახებ. ასეთ შემთხვევაში ამბობენ, რომ ადგილი აქვს დეკოდირებაზე უარის თქმას.

დავუშვათ, რომ არხით გადაიცემა $C(n,k,d)$ კოდის $\underline{c}$ სიტყვა, რომელიც ადიტიური ხმაურის ზემოქმედების შედეგად არხის გამოსასვლელზე გარდაიქმნება $\underline{r} \notin C$ ვექტორად, ანუ $\underline{r} = \underline{c} + \underline{f}$ ($\underline{c} \in C$, $\underline{f}$ - შეცდომის ვექტორია). დეკოდერი ახორციელებს მიღებული სიტყვის შეფასებას და გვაძლევს $\hat{c}$ კოდურ სიტყვას და ასევე განსაზღვრავს არხში არსებულ შესაძლო $\hat{f}$ ხმაურს, რასაც შეეძლო მივეყვანეთ $\underline{r}$ სიტყვამდე, ანუ

$$\underline{c} + \underline{f} = \underline{r} = \hat{\underline{c}} + \hat{\underline{f}}.$$

აქედან გამომდინარე, სწორად დეკოდირებისას გვაქვს $\underline{c} = \hat{\underline{c}}$ ($\underline{f} = \hat{\underline{f}}$), ხოლო მცდარად დეკოდირებისას გვაქვს $\underline{c} \neq \hat{\underline{c}}$ ($\underline{f} \neq \hat{\underline{f}}$).

ქვემოთ წარმოდგენილია დეკოდირების რამდენიმე მეთოდი ორობითი სიმეტრიული არხისათვის, რომლებიც დაფუძნებულია ჰემინგის მანძილზე.

ა) შეცდომების გამჟღავნება. დეკოდირების ეს მეთოდი განსაზღვრავს, არის თუ არა მიღებული $\underline{r}$ სიტყვა ერთ-ერთი კოდური სიტყვა. ამგვარად, დეკოდერი უბრალოდ ამოწმებს $\underline{r} \in C$ პირობას. შეცდომები მჟღავნდება როდესაც $\underline{r} \notin C$. $\underline{r}$ სწორად იქნება დეკოდირებული თუ შეცდომის ვექტორი $\underline{f} = 0,$ ე. ი. მიღებული $\underline{r}$ ვექტორი არ შეიცავს შეცდომებს. შემთხვევას, როდესაც $\underline{f} \in C, \underline{f} \neq 0$ მივყავართ $\underline{r}$ -ის მცდარ დეკოდირებამდე.

ბ) მაქსიმალური დამაჯერებლობით დეკოდირება. ამ მეთოდით დეკოდირებისას გამოითვლება იმის ალბათობა, რომ მიღებული $\underline{r}$ სიტყვა შეესაბამება ერთ-ერთ $\hat{\underline{c}}$ კოდურ სიტყვას. აირჩევა ისეთი კოდური სიტყვა, რომელსაც აქვს უდიდესი ალბათობა იმისა, რომ არჩეული იქნას გადაცემულ კოდურ სიტყვად. ე. ი. ვპოულობთ ალბათობის მაქსიმალურ მნიშვნელობას ყველა შესაძლო $\underline{c}$ -სთვის:

$$p(\underline{r} \mid \hat{\underline{c}}) = \max_{\underline{c} \in C} p(\underline{r} \mid \underline{c}).$$

თუ რამდენიმე კოდურ სიტყვას აქვს ერთნაირი მაქსიმალური ალბათობა, მათ შორის ერთ-ერთის არჩევა ხდება შემთხვევითი წესით. ორობითი სიმეტრიული არხისათვის მაქსიმალური დამაჯერებლობის დეკოდერი ირჩევს ისეთ კოდურ სიტყვას, რომელიც დაშორებულია უმცირესი ჰემინგის მანძილით მიღებული $\underline{r}$ სიტყვიდან. მაქსიმალური დამაჯერებლობით დეკოდირების შედეგად ვიღებთ მხოლოდ სწორ ან მცდარ კოდურ სიტყვას, ე. ი. ადგილი არა აქვს დეკოდირებაზე უარის თქმას.

გ) სიმბოლოების მაქსიმალური აპოსტერიორული ალბათობების გამოყენებით დეკოდირება. ამ მეთოდით დეკოდირებისას მოწმდება კოდური სიტყვის თითოეული c_i სიმბოლო. ყველა მათგანისათვის გამოითვლება 0-ისა და 1-ის გადაცემის ალბათობა. ამგვარად, დეკოდირების პროცესში მიღებული სიტყვის ყველა ელემენტისათვის ხდება დამოუკიდებული გადაწყვეტილების მიღება. ყველა n ელემენტის დეკოდირების შემდეგ მიღებული $(\hat{c}_0, \hat{c}_1, ..., \hat{c}_n)$ ვექტორი, განსხვავებით მაქსიმალური დამაჯერებლობით დეკოდირებისა, შეიძლება იყოს და შეიძლება არც იყოს კოდური სიტყვა. როდესაც დეკოდირებული სიტყვა არ წარმოადგენს კოდურ სიტყვას, ხოლო თვითონ კოდი მოცემულია არასისტემატური ფორმით, მაშინ ადგილი აქვს დეკოდირებაზე უარის თქმას. სისტემატური კოდირების გამოყენებისას ჩვენ შეგვიძლია განვაგრძოთ პროცედურა და დეკოდირებული სიტყვიდან ამოვარჩიოთ მხოლოდ საინფორმაციო პოზიციებზე განლაგებული სიმბოლოები. ასეთ შემთხვევაში ჩვენ უარს აღარ ვამბობთ დეკოდირებაზე და გვექნება სწორი ან მცდარი შედეგი, იმის მიხედვით, თანხვდენილია თუ არა გადაცემული საინფორმაციო სიმბოლოები მიღებული საინფორმაციო სიმბოლოებისა.

დ) მინიმალური მანძილით შემოსაზღვრული დეკოდირება. ამ პროცედურით დეკოდირებისას გადაწყვეტილება მიიღება მხოლოდ იმ შემთხვევაში, როდესაც მიღებული $\underline{r}$ ვექტორი მოთავსებულია $\left\lfloor \frac{d-1}{2} \right\rfloor$ რადიუსის მქონე გამასწორებელი სფეროს შიგნით. ასეთი დეკოდირებისას ადგილი

აქვს სამივე შესაძლო შედეგს: სწორ დეკოდირებას, მცდარ დეკოდირებას და დეკოდირებაზე უარის თქმას. მინიმალური მანძილით შემოსაზღვრული დეკოდირებისას შეცდომების გასწორება მოხდება მხოლოდ ისეთ შემთხვევაში, როდესაც არსებობს $\underline{c}$ კოდური სიტყვა, რომლისთვისაც სრულდება პირობა:

$$dist(\underline{c},\underline{r}) \le e = \left\lfloor \frac{d-1}{2} \right\rfloor .$$

ამგვარად, მინიმალური მანძილით შემოსაზღვრული დეკოდირების წესის თანახმად, ყველა მიღებული ვექტორი, რომელიც არ არის მოთავსებული გამასწორებელი სფეროს შიგნით არ შეიძლება იქნას დეკოდირებული. ასეთ ვექტორთა რიცხვი ზუსტად ტოლია ჰემინგის საზღვრის შესაბამისი უტოლობის (იხ. თეორემა 3.1) მარჯვენა და მარცხენა გამოსახულებების სხვაობის.

ე) მინიმალური მანძილის ნახევრის გარეთ დეკოდირება. ასეთ შემთხვევაში ადგილი აქვს დეკოდირებისას გადაწყვეტილების მიღების მცდელობას, როდესაც $\underline{r}$ ვექტორი მოთავსებულია $\left\lfloor \frac{d-1}{2} \right\rfloor$ რადიუსის მქონე გამასწორებელი სფეროს გარეთ. დეკოდირების მეთოდი განსხვავდება მაქსიმალური დამაჯერებლობით დეკოდირების მეთოდისაგან. დეკოდირების სფეროს რადიუსი მეტია $\left\lfloor \frac{d-1}{2} \right\rfloor$ სიდიდეზე. აქედან გამომდინარე, სხვადასხვა კოდური სიტყვების გარშემო მოთავსებული სფეროები ნაწილობრივ ფარავენ ერთმანეთს. ცხადია, რომ შესაძლებელია დეკოდირების სამი-

ვე შედეგი: სწორი დეკოდირება, მცდარი დეკოდირება და დეკოდირებაზე უარის თქმა. მაგალითად, თუ მიღებული $\underline{r}$ ვექტორი მდებარეობს ორი კოდური სიტყვის გარშემო მოთავსებული სფეროების გადაფარვის უბანზე, ჩვენ შეგვიძლია $\underline{r}$ ვექტორი გავაიგივოთ ამ ორი სიტყვიდან ერთერთთან ან საერთოდ უარი ვთქვათ გადაწყვეტილების მიღებაზე.

3.6 მაგალითში წარმოდგენილი დეკოდირების ტექნიკა კოდისათვის განმეორებით წარმოადგენს მაქსიმალური დამაჯერებლობით დეკოდირებას. ნებისმიერი კოდისათვის მაქსიმალური დამაჯერებლობით დეკოდირებისას ყველა კოდური სიტყვა უნდა შედარდეს მიღებულ სიტყვას. საზოგადოდ, ეს შეიძლება განხორციელდეს სტანდარტული განლაგებით დეკოდირების მეშვეობით. ამ პროცედურის ასახსნელად თავდაპირველად შემოვიტანოთ კოდის მომიჯნავე კლასის ცნება.

წრფივი ბლოკური $C(n,k,d)$ კოდის მომიჯნავე კლასი განისაზღვრება როგორც $\underline{b} \in F_2^n$ ვექტორის C კოდის ყველა კოდურ სიტყვასთან შეკრებით მიღებულ სიტყვათა სიმრავლე. აქედან გამომდინარე:

$$\left[C\right]_{\underline{b}} = \left\{\underline{b} + C\right\} = \left\{\underline{b} + \underline{c}, \underline{c} \in C\right\}.$$

მომიჯნავე კლასებს აქვთ შემდეგი თვისებები:

ა) თითოეული მომიჯნავე კლასი შეიცავს ზუსტად 2^k ვექტორს, რაც შეესაბამება C კოდში კოდურ სიტყვათა რიცხვს.

ბ) არსებობენ ისეთი $\underline{b}_1, \underline{b}_2, ..., \underline{b}_{2^{n-k_{-1}}}$ ვექტორები, რომ $C \cup \{\underline{b}_1 + C\} \cup \{\underline{b}_2 + C\} \cup ... \cup \{\underline{b}_{2^{n-k_{-1}}} + C\}$ სიმრავლე შეიცავს ვექტორული სივრცის ყველა 2^n ვექტორს.

გ) ორი მომიჯნავე კლასი ან თანხვედრილია ან გადაუკვეთელი (მათი ნაწილობრივი გადაფარვა შეუძლებელია).

ცხადია, რომ C კოდის ყველა მომიჯნავე კლასის გენერირებისთვის არცერთი $\underline{b}_i$ ვექტორი არ უნდა იყოს C-ს ელემენტი. წინააღმდეგ შემთხვევაში, თუ $\underline{b}_i \in C$, ნებისმიერი ვექტორი $\{\underline{b}_i + C\}$ სიმრავლიდან წარმოადგენს C-ს ელე-მენტს.

მიღებული ვექტორის დეკოდირება მომიჯნავე კლასების გამოყენებით ხორციელდება შემდეგნაირად. მომიჯნავე კლასები წარმოდგენილია სტრიქონებად, თანაც თითოეულ სტრიქონში (ე. ი. მომიჯნავე კლასში) უმცირესი წონის ვექტორს უკავია პირველი სვეტი. თუ მომიჯნავე კლასში ორ ან მეტ ვექტორს აქვს ერთნაირი უმცირესი წონა, მათ შორის არჩევანი ხდება შემთხვევითი წესით. არჩეულ პირველ ვექტორს ეწოდება მომიჯნავე კლასის ლიდერი. მიღებული $\underline{r} = \underline{c} + \underline{f}$ ვექტორის დეკოდირების პროცედურა იწყება იმ მომიჯნავე კლასის მოძებნით, რომელშიც მოთავსებულია $\underline{r}$ ვექტორი. შემდეგ ამ კლასის ლიდერი (უმცირესი ჰემინგის წონით) განიხილება როგორც შეცდომის ვექტორი, ემატება $\underline{r}$ ვექტორს და საბოლოოდ ვიღებთ კოდურ ვექტორს. დეკოდირების ამ მეთოდში ჩადებულია ის აზრი, რომ უმცირესი წონის შეცდომის ვექტორი უფრო მეტად ალბათურია, ვიდრე სხვა შეცდომის ვექტორი. ცხადია, რომ სტანდარტული გან-

ლაგებით დეკოდირება წარმოადგენს მაქსიმალური დამაჯერებლობით დეკოდირებას. ამასთან, პრაქტიკულად ასეთი მეთოდით დეკოდირება შესაძლებელია მხოლოდ მოკლე კოდებისათვის.

მაგალითი 4.1. განვიხილოთ $C(n=4,k=2,d=2)$ კოდი, რომელიც შედგება ოთხი კოდური სიტყვისაგან. სტანდარტული განლაგება მოიცავს ოთხ მომიჯნავე კლასს:

$$\underline{b}=(0000):\ \{(0000),\quad (0011),\quad (1100),\quad (1111)\}$$
$$\underline{b}=(1000):\ \{(1000),\quad (1011),\quad (0100),\quad (0111)\}$$
$$\underline{b}=(0010):\ \{(0010),\quad (0001),\quad (1110),\quad (1101)\}$$
$$\underline{b}=(1001):\ \{(1001),\quad (1010),\quad (0101),\quad (0110)\}.$$

ამ მაგალითში გამოყენებული მომიჯნავე კლასების მაგენერირებელი მინიმალური წონის ლიდერები არ არის ერთადერთი. ეს მოსალოდნელი იყო, ვინაიდან კოდის მინიმალური მანძილი ტოლია ორის, ე. ი. კოდს არ შეუძლია შეცდომების გასწორება. მაგალითისათვის, თუ $\underline{b}=(0100)$ ვექტორს დავუმატებთ C კოდის სიტყვებს, მივიღებთ მომიჯნავე კლასს: $\{(0100)\ (0111)\ (1000)\ (1011)\}$. ამგვარად, $\underline{b}=(0100)$ ვექტორი აგენერირებს იმავე მომიჯნავე კლასს, რასაც $\underline{b}=(1000)$ ვექტორი.

მინიმალური მანძილით შემოსაზღვრული დეკოდირების გამოყენებისას ჩვენ შეგვიძლია გავასწოროთ ნებისმიერი მიღებული $\underline{r}$ ვექტორი, რომელიც ეკუთვნის მომიჯნავე კლასს $\left\lfloor \frac{d-1}{2} \right\rfloor$-ზე ნაკლები ან ტოლი წონის მქონე ერთად-

ერთი ლიდერით. ჰემინგის საზღვარი ჩვენ გვაძლევს ინფორმაციას, თუ რა სხვაობაა მინიმალური მანძილით შემოსაზღვრული დეკოდირებისა და მაქსიმალური დამაჯერებლობით დეკოდირების შეცდომების გამასწორებელ მახასიათებლებს შორის. სრულყოფილი კოდებისათვის თითოეულ მომიჯნავე კლასს შეესაბამება $\left\lfloor \frac{d-1}{2} \right\rfloor$-ზე ნაკლები ან ტოლი წონის ლიდერი და ასეთი კოდებისათვის დეკოდირების აღნიშნულ ორივე მეთოდს მივყავართ ერთნაირ შედეგამდე.

5. შეცდომით დეკოდირების ალბათობა

განსაზღვრება 5.1. კოდური სიტყვის (ბლოკის) შეცდომით დეკოდირების ალბათობა - P_{Block} ტოლია გადაცემული კოდური სიტყვის მიღებულ კოდურ სიტყვასთან შეუსაბამობის ალბათობის (ე. ი. ადგილი აქვს მცდარ დეკოდირებას ან დეკოდირებაზე უარის თქმას). ანალოგიურად, ორობითი სიმბოლოს (ბიტის) შეცდომით დეკოდირების ალბათობა - $P_{Bit} = p$ განსაზღვრავს გადაცემული ბიტის მიღებულ ბიტთან შეუსაბამობის ალბათობას.

შეცდომით დეკოდირების ალბათობის გამოთვლისათვის აუცილებელია დეკოდირების სამი შესაძლო შედეგის განხილვა: კოდურ სიტყვაში შეცდომები სწორადაა გასწორებული, არ არის გასწორებული და მცდარად არის გასწორებული. როგორც უკვე აღვნიშნეთ მცდარ დეკოდირებას ადგილი აქვს ისეთ შემთხვევაში, როდესაც გადაცემული კოდური სიტყვისათვის $dist(\underline{c} + \underline{f}, \underline{c}) > dist(\underline{c} + \underline{f}, \underline{b})$, სადაც $\underline{c}, \underline{b} \in C$,

ხოლო $\underline{f}$ წარმოადგენს შეცდომის ვექტორს. მაგალითისთვის, $\underline{f} \in C,\ \underline{f} \neq 0$ შემთხვევისას ხდება მცდარი დეკოდირება.

განვსაზღვროთ ბლოკის შეცდომით დეკოდირების ალბათობა ორობითი სიმეტრიული არხისათვის. ასეთ არხში n სიგრძის მქონე ორობითი კოდური სიტყვის გადაცემისას არხის მიერ ზუსტად t შეცდომის წარმოქმნის ალბათობა ტოლია:

$$p(t) = p^t (1-p)^{n-t}.$$

სულ გვაქვს t წონის $\binom{n}{t}$ ვექტორი. ამგვარად, კოდური სიტყვის n პოზიციაზე ნებისმიერი კონფიგურაციის t შეცდომის წარმოქმნის ალბათობა ტოლია:

$$\binom{n}{t} p(t) = \binom{n}{t} p^t (1-p)^{n-t}.$$

მინიმალური მანძილით შემოსაზღვრული დეკოდირებისას ბლოკზე შეცდომას ადგილი აქვს არხის მიერ e-ზე მეტი შეცდომის წარმოქმნის შემთხვევაში, ანუ

$$P_{Block} = \sum_{j=e+1}^{n} \binom{n}{j} p^j (1-p)^{n-j}.$$

ამგვარად, დეკოდირების ამ მეთოდით გარანტირებულად შესაძლებელია $e = \left\lfloor \frac{d-1}{2} \right\rfloor$-ის ტოლი ან ნაკლები შეცდომის გასწორება. უნდა აღინიშნოს, რომ მინიმალური მანძილით შემოსაზღვრული დეკოდირების მეთოდით მრავალ შემთხვევაში შეიძლება გასწორდეს უფრო მეტი შეცდომაც. ამი-

ტომ, ზემოთ მოყვანილი ფორმულა წარმოგვიდგენს ბლოკზე შეცდომის ალბათობის ზედა საზღვარს. ამასთან, იგი შეიძლება ჩაიწეროს ეკვივალენტური ფორმითაც:

$$P_{Block} = 1 - \sum_{j=0}^{e} \binom{n}{j} p^j (1-p)^{n-j}.$$

განვიხილოთ ახლა შემთხვევა, როდესაც გვაინტერესებს მხოლოდ შეცდომების გამჟღავნება. დეკოდირებისას შეცდომა წარმოიშობა მაშინ, როდესაც არხში არსებული შეცდომებს გადაცემული კოდური სიტყვა გადაჰყავთ სხვა კოდურ სიტყვაში. კოდის წონათა აღნუსხვის (განაწილების) $W = (w_0, w_1, ...w_n)$ ვექტორის გამოყენებით ვიღებთ შეცდომების გაუმჟღავნელობის P_{EBlock} ალბათობას:

$$P_{EBlock} = \sum_{j=1}^{n} w_j p^j (1-p)^{n-j} \le 2^{-(n-k)}.$$

შესაბამისად, არხისათვის ყველაზე კრიტიკულ ($p = \frac{1}{2}$) შემთხვევაში მივიღებთ:

$$P_{EBlock} = \sum_{j=1}^{n} w_j \left(\frac{1}{2}\right)^n = \left(\frac{1}{2}\right)^n \sum_{j=1}^{n} w_j = \left(\frac{1}{2}\right)^n (2^k - 1) \approx 2^{-(n-k)}.$$

თუ $\underline{f}$ წარმოადგენს F_2^n ვექტორული სივრციდან შემთხვევით არჩეულ ვექტორს, მაშინ შეცდომების გაუმჟღავნებლობის P_{EBlock} ალბათობა ტოლია $\underline{f}$-ის არანულოვან კოდურ სიტყვასთან თანხვდენილობის ალბათობის.

<u>მაგალითი 5.1.</u> $n = 3$ სიგრძის მქონე კოდს განმეორებით შეუძლია ერთეულოვანი შეცდომის გასწორება. ამიტომ p შეცდომის ალბათობის შემცველი ორობითი სიმეტრიული

არხით კოდური სიტყვების გადაცემისას ბლოკის შეცდომით დეკოდირების ალბათობისათვის გვექნება:

$$P_{Block} = 1 - \sum_{j=0}^{1} \binom{3}{j} p^j (1-p)^{3-j} = 1 - (1-p)^3 - 3p(1-p)^2.$$

<u>მაგალითი 5.2.</u> განვიხილოთ $n = 3$ სიგრძის მქონე ლუწობის შემმოწმებელი კოდის წონათა აღნუსხვის $W = (1, 0, 3, 0)$ ვექტორი. შეცდომების გაუმჟღავნებლობის ალბათობა ამ შემთხვევისათვის ტოლია:

$$P_{EBlock} = \sum_{j=1}^{3} w_j p^j (1-p)^{n-j} = 3p^2(1-p).$$

მაქსიმალური დამაჯერებლობით დეკოდირებისას ბლოკზე შეცდომის ალბათობა ტოლია:

$$P_{MBlock} = 1 - \sum_{j=0}^{n} \alpha_j p^i (1-p)^{n-j}, \quad \alpha_0 = 1,$$

სადაც α_j მომიჯნავე კლასების j წონის მქონე ლიდერთა საერთო რიცხვია. α_j -ს განაწილება შეიძლება განისაზღვროს მხოლოდ მოკლე კოდებისათვის. რაც შეეხება გრძელ კოდებს, მაქსიმალური დამაჯერებლობით დეკოდირების პრაქტიკული რეალიზაცია მათთვის წარმოადგენს საკმაოდ რთულ ამოცანას.

<u>მაგალითი 5.3.</u> ბლოკური (15, 7, 5) კოდისათვის ჩვენ შეგვიძლია ვიანგარიშოთ არხში შეცდომის ალბათობა p და დეკოდირების სხვადასხვა მეთოდისათვის ავაგოთ ბლოკის დეკოდირების მახასიათებლების p -ზე დამოკიდებულების გრაფიკები. წონათა აღრიცხვის განაწილება ამ კოდისათვის შეიძლება ჩაიწეროს მრავალწევრის სახით:

$$W(x) = 1 + 18x^5 + 30x^6 + 15x^7 + 15x^8 + 30x^9 + 18x^{10} + x^{15},$$

ანუ კოდი ნულოვანი სიტყვის გარდა შეიცავს 18 სიტყვას წონით 5, 30 სიტყვას წონით 6 და ა. შ. გამოთვლის შედეგები მოყვანილია მე-3 ნახაზზე, თანაც მახასიათებლების უფრო თვალნათლივ წარმოდგენისათვის მარჯვენა და მარცხენა დიაგრამებზე გამოყენებულია P_{Block} ორდინატთა ღერძის განსხვავებული მასშტაბი.

6. ჰემინგის კოდები

განსაზღვრება 6.1. ჰემინგის კოდის ლუწობის შემოწმებული H მატრიცა შეიცავს F_2^h ვექტორული სივრცის ერთმანეთისაგან განსხვავებულ $2^h - 1$ არანულოვან სვეტს.

თეორემა 6.1. ორობითი ჰემინგის კოდის პარამეტრებია: სიგრძე: $n = 2^h - 1,$ განზომილება: $k = n - h,$ მინიმალური მანძილი: $d = 3.$

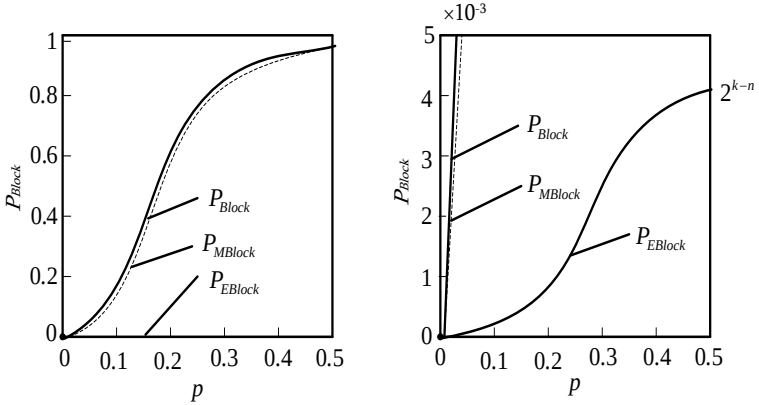
დამტკიცება: n და k სიდიდეების მნიშვნელობა გამომდინარეობს შემმოწმებელი მატრიცის ფორმიდან. ამასთან H მატრიცის ნებისმიერი ორი სვეტი წრფივად დამოუკიდებელია და არსებობს წრფივად დამოკიდებული სამი სვეტი. აღნიშნულიდან გამომდინარე, მინიმალური მანძილი $d = 3.$ □

ჰემინგის კოდი ასწორებს ნებისმიერ ერთეულოვან შეცდომას და ამჟღავნებს ყველა ლუწი რაოდენობის შეცდომას.

თეორემა 6.2. ჰემინგის კოდი სრულყოფილი კოდია.

დამტკიცება: შევიტანოთ ჰემინგის კოდის პარამეტრები $n = 2^h - 1, \quad k = n - h, \; e = 1 = \left\lfloor \frac{d-1}{2} \right\rfloor$ 3.1 თეორემაში. გვექნება, $2^k + 2^k(1 + \binom{n}{1}) \le 2^n, \; 1 + n \le 2^{n-k}$ უტოლობები და საბოლოოდ $1 + 2^h - 1 \le 2^h$ ტოლობა. ამგვარად, ყველა ჰემინგის კოდი სრულყოფილი კოდია. □

ავაგოთ კონკრეტული ჰემინგის კოდი და მოვახდინოთ მისი დეკოდირება.



ნახ. 3. შეცდომის ალბათობის მახასიათებლების შედარება შეცდომების გამჟღავნების, შემოსაზღვრული მანძილით და მაქსიმალური დამაჯერებლობის მეთოდებით დეკოდირებისას

<u>მაგალითი 6.1.</u> ავაგოთ ორობითი ჰემინგის კოდი $h = 3$ შემთხვევისათვის. ლუწობის შემმოწმებელი H მატრიცა შეიცავს $n = 2^3 - 1 = 7$ სვეტს და $n - k = h = 3$ სტრიქონს. ჩვენ შეგვიძლია H მატრიცას მივცეთ ისეთი სახე, სადაც სვეტის

რიგითი ნომრების ორობითი ფორმით წარმოდგენა განსაზღვრავს სვეტის ელემენტებს:

$$H = \begin{pmatrix} 0001111 \\ 0110011 \\ 1010101 \end{pmatrix}, \qquad H\underline{c}^T = 0, \quad \underline{c} \in C .$$

3-ის ტოლი მინიმალური მანძილისათვის ნებისმიერი ორი სვეტი უნდა იყოს წრფივად დამოუკიდებელი. ეს პირობა სრულდება, ვინაიდან H მატრიცის ნებისმიერი ორი სვეტი განსხვავდება ერთმანეთისაგან. ამასთან არსებობს წრფივად დამოკიდებული სამი სვეტი, მაგალითად, პირველი, მეორე და მესამე სვეტები. ამგვარად, კოდის მინიმალური მანძილი $d = 3$. მაგალითისათვის, $\underline{c} = (111000)$ მინიმალური წონის მქონე კოდური სიტყვაა, ვინაიდან:

$$H\underline{c}^T = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} = \underline{0}.$$

განვიხილოთ შემთხვევა, როდესაც აღნიშნული $\underline{c}$ ვექტორის გადაცემისას არხში ადგილი აქვს ერთეულოვან შეცდომას, ანუ შეცდომის ვექტორი $\underline{f} = (0000010)$, ხოლო მიღებული ვექტორი $\underline{r} = \underline{c} + \underline{f} = (111010)$. 3.11 განსაზღვრების შესაბამისად ტრანსპონირებული სინდრომის ვექტორი ტოლია:

$$\underline{s}^T = H\underline{r}^T = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} = H\underline{f}^T .$$

შესაბამისად, დეკოდირების შედეგი გვაძლევს სინდრომს, რომელიც გვიჩვენებს კოდურ სიტყვაში შეცდომით მიღებული ბიტის ნომერს ორობით ფორმაში.

ახლა წარმოვადგინოთ 6.1 მაგალითში მოყვანილი ჰემინგის კოდი სისტემატური ფორმით.

<u>მაგალითი 6.2.</u> H მატრიცის სტრიქონების წრფივი კომბინაციით ჩვენ შეგვიძლია ავაგოთ $n = 7$ სიგრძის მქონე ორობითი ჰემინგის კოდის ლუწობის შემმოწმებელი მატრიცა სისტემატური კოდირებისათვის:

$$H = \left(\begin{array}{cccc|ccc} 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{array}\right).$$

7. გილბერტ-ვარშამოვის საზღვარი

როგორც უკვე აღვნიშნეთ კოდის მოცემული n და k პარამეტრებისათვის ჰემინგის საზღვარი გვიჩვენებს კოდის მინიმალური მანძილის ზედა საზღვარს. მისგან განსხვავებით გილბერტ-ვარშამოვის ქვედა საზღვარი მიგვითითებს n სიგრძისა და k განზომილების კოდების არსებობის შესახებ გარკვეული d მინიმალური მანძილით. განასხვავებენ კოდის მინიმალური მანძილის ორ ქვედა საზღვარს, რომლებიც ერთმანეთისაგან დამოუკიდებლად იქნა დამტკიცებული ე. გილბერტისა და რ. ვარშამოვის მიერ. ამასთან ვარშამოვის საზღვარი მართებულია მხოლოდ წრფივი ბლოკური კოდებისათვის, მაშინ როდესაც გილბერტის საზღვარი მოიცავს არაწრფივი კოდების შემთხვევასაც. ქვემოთ ჩვენ შევჩერდე-

ბით ვარშამოვის საზღვარზე, ვინაიდან იგი იძლევა მანძილის უკეთეს საზღვარს, ვიდრე გილბერტის საზღვარი.

თეორემა 7.1. არსებობს n სიგრძის, k განზომილების და არანაკლებ d მინიმალური მანძილის მქონე კოდი, რომლისთვისაც სრულდება უტოლობა:

$$\sum_{j=0}^{d-2}\binom{n}{j} \geq 2^{n-k}.$$

დამტკიცება: ორობით წრფივ კოდს აქვს d -ს ტოლი მინიმალური მანძილი თუ ლუწობის შემმოწმებელი H მატრიცის $d-1$ სვეტი წრფივად დამოუკიდებულია. ავაგოთ ლუწობის შემმოწმებელი H მატრიცა, რომელიც შეიცავს $n-k$ სტრიქონს და n სვეტს. მატრიცის პირველი სვეტად შეიძლება ავირჩიოთ $n-k$ სიგრძის მქონე ნებისმიერი არანულოვანი ვექტორი. შემდეგ ავირჩიოთ $n-k$ სიგრძის მქონე არანულოვანი ვექტორი, რომელიც განსხვავდება პირველ სვეტში არჩეული ვექტორისაგან და განვიხილოთ ის, როგორც H მატრიცის მეორე სვეტი. მატრიცის მესამე სვეტი შეიძლება იყოს $n-k$ სიგრძის მქონე ნებისმიერი არანულოვანი ვექტორი, რომელიც არ წარმოადგენს პირველი ორი სვეტის წრფივ კომბინაციას. საზოგადოდ, H მატრიცის d -ურ სვეტად აირჩევა $n-k$ სიგრძის მქონე ნებისმიერი არანულოვანი ვექტორი, რომელიც არ წარმოადგენს H მატრიცის $d-2$ ან ნაკლები რაოდენობის უკვე არჩეული სვეტების წრფივ კომბინაციას. შემმოწმებელი მატრიცის ასეთი წესით აგებისას, შეიძლება დარწმუნებული ვიყოთ, რომ მატრიცის $d-1$ ან ნაკლები რაოდენობის სვეტების არცერთი წრფივი კომბინაცია არ მოგვცემს ნულოვან ვექტორს. H მატრიცისათვის

შემდეგი სვეტის დამატება შესაძლებელია მხოლოდ იქამდე, სანამ მატრიცის უკვე არჩეული $d-2$ ან ნაკლები რაოდენობის სვეტების ყველა წრფივი კომბინაციების ერთობლიობა არ ამოწურავს $n-k$ სიგრძის ყველა შესაძლო არანულოვან ვექტორთა სიმრავლეს. ამგვარად, ვიდრე

$$\binom{j-1}{1}+\binom{j-1}{2}+\ldots+\binom{j-1}{d-2}$$

სიდიდე ნაკლებია $n-k$ სიგრძის მქონე ნულისაგან განსხვავებულ ვექტორთა $2^{n-k}-1$ რიცხვზე, აუცილებლად მოიძებნება კიდევ ერთი სვეტი, რომელიც შეიძლება მიუერთდეს H მატრიცას. მაშასადამე, თუ სრულდება პირობა:

$$\binom{j-1}{1}+\binom{j-1}{2}+\ldots+\binom{j-1}{d-2}<2^{n-k}-1,$$

მაშინ არსებობს j სიგრძისა და არა უმცირეს $j-(n-k)$ განზომილების მქონე კოდი, რომლის მინიმალური მანძილი ტოლია d-სი. აღვნიშნოთ n-ით j-ს უდიდესი მნიშვნელობა, რომლისთვისაც სრულდება ეს უტოლობა. მაშინ არსებობს (n,k,d)-კოდი, რომლის მინიმალური მანძილიც აკმაყოფილებს შემდეგ უტოლობას:

$$\binom{n}{1}+\binom{n}{2}+\ldots.+\binom{n}{d-2}\geq 2^{n-k}-1. \qquad \square$$

8. წარმომქმნელი მატრიცა

ახლა განვიხილოთ, თუ როგორ შეიძლება აიგოს $\underline{i}=(i_0,i_1,\ldots.,i_{k-1})$ საინფორმაციო ვექტორის მიხედვით

$\underline{c} = (c_0, c_1, ..., c_{n-1})$ კოდური სიტყვა. ვინაიდან წრფივი კოდი განისაზღვრება როგორც წრფივი ვექტორული სივრცე, უნდა არსებობდეს ამ სივრცის ბაზისი. G მატრიცის შემადგენელი k წრფივად დამოკიდებული ვექტორი წარმოქმნის ვექტორული სივრცის ბაზისს. G მატრიცას ეწოდება კოდის წარმომქმნელი მატრიცა და თითოეული კოდური $\underline{c}$ სიტყვა გამოითვლება $\underline{c} = \underline{i}\, G$ განტოლების საფუძველზე.

წარმომქმნელი მატრიცა $k \times n$ ზომების მატრიცაა. მოცემული წარმომქმნელი G მატრიცის მიხედვით მის სტრიქონებზე ელემენტარული ოპერაციების ჩატარებით (ე. ი. მათი გადანაცვლებით და შეკრებით) ჩვენ შეგვიძლია ავაგოთ G' წარმომქმნელი მატრიცა. ახალი მატრიცა ახდენს იმავე კოდის გენერირებას, ოღონდაც სტრიქონების გადანაცვლების შედეგად იცვლება საინფორმაციო ვექტორების კოდურ სიტყვაში ასახვის წესი.

განვსაზღვროთ H ლუწობის შემმოწმებელი და G წარმომქმნელი მატრიცების ფორმა კოდირების სისტემატური მეთოდის გამოყენების შემთხვევაში.

$c_0 = i_0,\ c_1 = c_1,\ \ldots,\ c_{k-1} = i_{k-1}$ ტოლობებიდან გამომდინარეობს შემდეგი გამოსახულებები:

$$H\underline{c}^T = \left(A \mid I\right)\underline{c}^T = \underline{0}, \quad \begin{pmatrix} c_k \\ \vdots \\ c_{n-1} \end{pmatrix} = -A \begin{pmatrix} c_0 \\ \vdots \\ c_{k-1} \end{pmatrix} = -A \begin{pmatrix} i_0 \\ \vdots \\ i_{k-1} \end{pmatrix},$$

$$\underline{c}^T = \begin{pmatrix} I' \\ \hline -A \end{pmatrix} \underline{i}^T, \quad \underline{c} = \underline{i}(I' \mid -A^T) = \underline{i}G,$$

სადაც I შეესაბამება $(n-k)\times(n-k)$ ზომების, ხოლო $I'-k\times k$ ზომების ერთეულოვან მატრიცას.

<u>მაგალითი 8.1.</u> 6.2 მაგალითში მოცემული ჰემინგის კოდის ლუწობის შემმოწმებელი

$$H=\left(\begin{array}{cccc|ccc}0&1&1&1&1&0&0\\1&0&1&1&0&1&0\\1&1&0&1&0&0&1\end{array}\right)=(A\,|\,I)$$

მატრიცის მიხედვით ავაგოთ წარმომქმნელი მატრიცა:

$$G=(I'\,|\,-A^T)=\left(\begin{array}{cccc|ccc}1&0&0&0&0&1&1\\0&1&0&0&1&0&1\\0&0&1&0&1&1&0\\0&0&0&1&1&1&1\end{array}\right).$$

წარმომქმნელი მატრიცის დახმარებით ვპოულობთ, რომ, მაგალითად, $\underline{i}=(1110)$ საინფორმაციო ვექტორს შეესაბამება

$$\underline{c}=\underline{i}G=(1000011)+(0100101)+(0010110)=(1110000)$$

კოდური სიტყვა.

9. კოდირება, ციკლური კოდები, დუალური კოდები, კოდების დამოკლება და დაგრძელება

კოდი წარმოადგენს კოდური სიტყვების სიმრავლეს. კოდირება ეს არის საინფორმაციო ვექტორების სიმრავლის ასახვის მეთოდი კოდურ სიტყვათა სიმრავლეში.

საზოგადოდ, ორობითი კოდისათვის არსებობს ასეთი ასახვის $2^k!$ განსხვავებული შესაძლებლობა: პირველი საინფორმაციო ვექტორისათვის ჩვენ შეგვიძლია ავირჩიოთ 2^k შესაძლებელი კოდური სიტყვიდან ერთ-ერთი, მეორე საინფორმაციო ვექტორისათვის - ერთ-ერთი დარჩენილი 2^k-1 კოდური სიტყვიდან, და ა. შ. რა თქმა უნდა, ასეთ შემთხვევაში ჩვენ ვუშვებთ, რომ ნულებისაგან შემდგარი საინფორმაციო ვექტორი შეიძლება აისახოს არანულოვან კოდურ სიტყვაში.

ყველა შესაძლო ასახვის მეთოდს შორის არსებობს სისტემატური ასახვის პროცედურები. ცხადია, რომ ყველა ისეთ ასახვის პროცედურას, რომლის დროსაც კოდური სიტყვის შესაბამისი საინფორმაციო ვექტორის შემადგენელი სიმბოლოები შეუცვლელად თავსდება კოდური სიტყვის ფიქსირებულ k პოზიციაზე, ეწოდება სისტემატური. ყველა წრფივი ბლოკური კოდი შეიძლება კოდირებულ იქნას სისტემატურ ფორმაში.

მოცემული წარმომქმნელი G მატრიცა განსაზღვრავს საინფორმაციო ვექტორების კოდურ სიტყვაში ასახვის გარკვეულ წესს. ცხადია, რომ თუ ჩვენ G მატრიციდან მის სტრიქონებზე ელემენტარული ოპერაციების ჩატარებით მივიღებთ ახალ G' მატრიცას, ჩვენ შევცვლით ასახვის წესს, მაგრამ შევინარჩუნებთ კოდური სიტყვების იმავე სიმრავლეს და, აქედან გამომდინარე, იმავე კოდს.

წარმომქმნელი G მატრიცის სვეტების გადანაცვლებით მიიღება კოდური სიტყვების განსხვავებული სიმრავლე და, ე. ი. ახალი კოდი. მიუხედავად ამისა, კოდის ისეთი თვი-

სებები როგორიცაა სიჩქარე, მინიმალური მანძილი და წონების განაწილება ორივე კოდისათვის ერთნაირია და, ამიტომ, ახალ კოდს ეწოდება საწყისი კოდის ეკვივალენტური კოდი. შემოვიტანოთ ახლა ციკლური და დუალური კოდების ცნება.

$\underline{c} = (c_0, c_1, ..., c_{n-1})$ კოდური სიტყვის ციკლური წანაცვლება i კოორდინატის მიხედვით გვაძლევს $(c_i, c_{i+1}, ..., c_{n-1}, c_0, c_1, ..., c_{i-1})$ ვექტორს.

განსაზღვრება 9.1. კოდს ეწოდება ციკლური თუ მისი კოდური სიტყვების კოორდინატების ყველა ციკლური წანაცვლება ისევ გვაძლევს კოდურ სიტყვას.

ციკლური კოდები შეიძლება წარმოდგენილი იქნას მრავალწევრების მეშვეობით და ისინი საინტერესოა პრაქტიკული თვალსაზრისით. ეს გამოწვეულია იმით, რომ მრავალწევრების შეკრება-გამრავლების ოპერაციები შეიძლება მარტივად იქნას რეალიზებული წანაცვლების რეგისტრების გამოყენებით.

კოდის წარმომქმნელი მატრიცა შეიძლება გამოყენებულ იქნას ამ კოდთან დაკავშირებული კოდის ლუწობის შემმოწმებელ მატრიცად და პირიქით. ასეთი პრინციპით დაკავშირებული კოდებს უწოდებენ დუალურ კოდებს.

განსაზღვრება 9.2. H ლუწობის შემმოწმებელი მატრიცის მქონე C კოდის დუალური C^1 კოდი, ეს არის კოდი, რომლის წარმომქმნელი G^1 მატრიცა ტოლია H -ის.

გამოვიყენოთ 3.4 განსაზღვრებით მოცემული სკალარული ნამრავლი დუალური კოდებისათვის. გვაქვს,

$$\underline{c} \in C, \quad \underline{b} \in C^1 \mid \left\langle \underline{c}\underline{b}^T \right\rangle = \underline{0}.$$

ანუ, ვინაიდან ლუწობის შემმოწმებელი მატრიცის თითოეული (ტრანსპონირებული) სტრიქონის ნამრავლი წარმომქმნელ მატრიცაზე უნდა გვაძლევდეს ნულოვან ვექტორს, იგივე სამართლიანი უნდა იყოს აღნიშნული სტრიქონების ყველა წრფივი კომბინაციისთვისაც.

დუალური C^1 კოდის სიგრძე ტოლია C კოდის სიგრძის, $n^1 = n$. C^1 კოდის განზომილება გამოითვლება შემდეგნაირად, $k^1 = n - k$. აღვნიშნოთ, რომ არ არსებობს C^1 კოდის მინიმალური მანძილის გამოთვლის მეთოდი უშუალოდ C კოდის მიხედვით.

<u>მაგალითი 9.1.</u> 8.1 მაგალითში მოცემული ჰემინგის კოდის დუალური კოდის წარმომქმნელ და ლუწობის შემმოწმებელ მატრიცებს აქვთ შემდეგი სახე:

$$G^1 = H = \left(\begin{array}{cccc|ccc} 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{array}\right) = (A \mid I),$$

$$H^1 = G = \left(\begin{array}{cccc|ccc} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{array}\right).$$

H მატრიცის პირველი სტრიქონის სკალარული ნამრავლი G მატრიცის მეორე სტრიქონზე გვაძლევს: $0 \cdot 0 + 1 \cdot 1 + 1 \cdot 0 + 1 \cdot 0 + 1 \cdot 1 + 0 \cdot 0 + 0 \cdot 1 = 2 = 0 \bmod 2.$

კოდების დამოკლება შეიძლება განხორციელდეს შემდეგი ორი მეთოდის გამოყენებით:

ა) k სიმბოლოსაგან შემდგარი საინფორმაციო $\underline{i}$ ვექტორის ნაცვლად ჩვენ ვირჩევთ მის მხოლოდ პირველ $k-m$ სიმბოლოს. ამგვარად, როგორც კოდის განზომილება, ასევე მისი სიგრძე შემცირებულია m სიდიდით, ხოლო კოდის სიჩქარე ტოლია $\frac{k-m}{n-m}$. ამასთან ახალი კოდის მინიმალური მანძილი მეტი ან ტოლია საწყისი კოდის მინიმალურ მანძილისა.

ბ) ვახდენთ თითოეული კოდური სიტყვის პერფორაციას, ანუ ფიქსირებული m კოორდინატის შესაბამისი სიმბოლოები ამოგდებულები არიან კოდური სიტყვებიდან. პერფორირების შედეგად მიღებული კოდის სიჩქარე ტოლია $\frac{k}{n-m}$ სიდიდის, ხოლო მისი მინიმალური მანძილი ნაკლები ან ტოლია საწყისი კოდის მინიმალური მანძილისა.

წრფივი კოდების დამოკლების ორივე მეთოდი გვაძლევს წრფივ კოდებს.

<u>მაგალითი 9.2</u>. დავუბრუნდეთ 8.1 მაგალითში მოცემული კოდის წარმომქმნელ მატრიცას. გამოვიყენოთ კოდის დამოკლების (ა) მეთოდი: გავანულოთ საინფორმაციო ვექტორის პირველი ორი სიმბოლო და შემდეგ ამოვაგდოთ ეს სიმბოლოები კოდური სიტყვიდან. ამგვარად, 8.1 მაგალითში განხილული წარმომქმნელი მატრიცის გამოყენებით, $\underline{i}=(0010)$ საინფორმაციო ვექტორისთვის მივიღებთ:

$$\underline{c}=\underline{i}G=(0010110)\Rightarrow(10110).$$

შესაბამისად, კოდური სიტყვის სიგრძეა $n=5$. ახლა გამოვიყენოთ კოდის დამოკლების (ბ) მეთოდი: ამოვაგდოთ, მა-

გალითად ყველა კოდური სიტყვის მე-3 ელემენტი, ე. ი. $(0010110) \Rightarrow (000110)$. ამგვარად, კოდის სიჩქარე იზრდება 4/7 სიდიდიდან 4/6 = 2/3 სიდიდემდე.

კოდის დაგრძელებისათვის საკმარისია თითოეულ კოდურ სიტყვას დავუმატოთ ერთი სიმბოლო (0 ან 1) ისე, რომ მიღებულ კოდურ სიტყვას ჰქონდეს ჰემინგის ლუწი წონა. ასეთი ტექნიკის გამოყენება მიზანშეწონილია მხოლოდ ისეთი კოდებისათვის, რომელთაც აქვთ ჰემინგის კენტი მინიმალური წონა. მათი მინიმალური მანძილი იზრდება ერთის ტოლი სიდიდით.

თეორემა 9.1. განვიხილოთ $C(n,k,d)$ კოდი, სადაც d კენტია, $(n-k)\times n$ ზომების მქონე ლუწობის შემმოწმებელი H მატრიცით. ჩვენ შეგვიძლია ავაგოთ ლუწობის შემმოწმებელი მატრიცა $(n-k+1)\times(n+1)$ ზომებით, თუ H -ს შევავსებთ ერთიანებისაგან შემდგარი სტრიქონითა და $(1000...0)^T$ ვექტორ-სვეტით. ახალი დაგრძელებული კოდის პარამეტრებია $(n+1,k,d+1)$, ხოლო მის ლუწობის შემმოწმებელ მატრიცას აქვს სახე:

$$H^{ext} = \begin{pmatrix} 1 & 1 & \dots & 1 & 1 \\ & & & & 0 \\ & H & & & \vdots \\ & & & & 0 \end{pmatrix}.$$

დამტკიცება. კოდის სიგრძე და განზომილება განისაზღვრება H^{ext} მატრიცის პარამეტრებით. ვინაიდან კოდი წრფივია, მისი მინიმალური მანძილი ტოლია უმცირესი წონის კოდური სიტყვის წონის. ავიღოთ $\underline{c} \in C$, რომლის-

თვისაც $wt(\underline{c}) = d$. ამ სიტყვიდან დაგრძელებულ $\underline{c}^{ext}$ კოდურ სიტყვაზე გადასვლისას წონა იზრდება $wt(\underline{c}^{ext}) = d + 1$ სიდიდემდე. □

<u>მაგალითი 9.3.</u> ისევ განვიხილოთ H ლუწობის შემმოწმებელი მატრიცა 8.1 მაგალითიდან. დაგრძელებული კოდის ლუწობის შემმოწმებელი მატრიცა ტოლია:

$$H = \left(\begin{array}{cccc|cccc} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \end{array}\right).$$

მაგალითისათვის, $\underline{c} = (00101101)$ კოდური სიტყვა გადადის $\underline{c}^{ext} = (00101101)$ კოდურ სიტყვაში.

10. რიდ-სოლომონის კოდები

რიდ-სოლომონის (RS) კოდები განეკუთვნება პრაქტიკულ სისტემებში ხშირად გამოყენებული კოდების კლასს. ჩვენ დეტალურად შევისწავლით RS კოდების თვისებებს. კოდების წარმოდგენის პროცესში გამოყენებული იქნება ინჟინრული მიდგომისათვის დამახასიათებელი გარდაქმნის ტექნიკა, რაც გაგვიიოლებს მათი მახასიათებლების ანალიზს. ცენტრალურ ნაწილს შეადგენს RS კოდების დეკოდირების ალგორითმი, რომლის მიხედვითაც შეიძლება გადავჭრათ ისეთი პრობლემა, როგორიცაა არხიდან მიღებული კოდური სიტყვიდან გადაცემული სიტყვის შესახებ ინფორმაციის ამო-

ღება. დეკოდირების მეთოდის რეალიზაციისას საჭირო ხდება ე. წ. „გასაღები“ განტოლების ამოხსნა, რომელიც ქვემოთ გადაწყვეტილია მესი-ბერლეკამპის ალგორითმის გამოყენებით.

ჩვენ განვიხილავთ მხოლოდ მარტივ რიცხვებზე დაფუძნებულ გალუას ველებს GF (იხ. მაგალითად [1-4], [8]), თუმცა ქვემოთ მოყვანილი ყველა არგუმენტი და ალგორითმი შეიძლება მისადაგებულ იქნას ველების გაფართოებისთვისაც.

გამოვიყენოთ გალუას ველებისათვის ალგებრის შემდეგი ფუნდამენტური თეორემა.

თეორემა 10.1. $A(x) = A_0 + A_1 x + A_2 x^2 + ... + A_{k-1} x^{k-1}$ მრავალწევრს, რომლის ხარისხია $k-1$ $(A_{k-1} \neq 0)$, ხოლო კოეფიციენტებია $A_i \in GF(p)$, აქვს არა უმეტესი $k-1$ რაოდენობის $\alpha_j \in GF(p)$ ფესვი.

დამტკიცება. თუ $\alpha \in GF(p)$ $A(x)$-ის ფესვია, მაშინ $(x-\alpha)$ წარმოადგენს ამ მრავალწევრის წრფივ მამრავლს, ანუ $(x-\alpha) \mid A(x)$. შესაბამისად, $A(x) = (x-\alpha)A^*(x)$, სადაც მრავალწევრთა ხარისხისათვის (deg) ადგილი აქვს ტოლობას $\deg(A(x)) = \deg(A^*(x)) + 1$. აღნიშნული თვისებით შეგვიძლია ვისარგებლოთ $\deg(A(x))$-ჯერ, რაც გვაძლევს $A(x)$-ის არაუმეტეს $k-1$ წრფივ მამრავლს. □

ახლა, ვნახოთ, თუ როგორ შეიძლება ამ მრავალწევრის გამოყენება.

თეორემა 10.2. განვიხილოთ $GF(p)$ გალუას ველის n განსხვავებული არანულოვანი $(a_0, a_1, ..., a_{n-1})$ ელემენტი, სა-

დაც $n \le p-1$. დავუშვათ, რომ $A(x)$ მრავალწევრის ხარისხი ტოლია $(k-1)$-ის, ხოლო მისი კოეფიციენტები მოთავსებულია $GF(p)$-ში. ასევე დავუშვათ, რომ $k-1 \le n-d$, სადაც d მოცემული მთელი რიცხვია. განვიხილოთ $\underline{a} = (a_0, a_1, ..., a_{n-1})$ ვექტორი, რომლის ელემენტებია $a_i = A(\alpha_i),\ i = 0,1,...,n-1.$ $\underline{a}$ ვექტორის წონა მეტი ან ტოლია d-ზე, ე. ი. $wt(\underline{a}) \ge d.$

დამტკიცება. 10.1 თეორემის მიხედვით $A(x)$-ს აქვს არა უმეტეს $k-1$ ფესვისა. აქედან გამომდინარე, $\underline{a}$ შეიცავს სულ ცოტა $n-k+1 \ge d$ ელემენტს, რომლებიც არ წარმოადგენენ $A(x)$-ის ფესვებს. □

ამ თეორემის გამოყენებით ჩვენ შეგვიძლია ავაგოთ გარკვეულ ფიქსირებულ სიდიდეზე მეტი წონის ვექტორი, თანაც ეს სიდიდე განსაზღვრავს კოდის მინიმალურ წონას და მინიმალურ მანძილს.

განსაზღვრება 10.1. დავუშვათ, რომ $\alpha \in GF(p)$ n რიგის ელემენტია. n სიგრძის RS კოდი განისაზღვრება k სიდიდეზე ნაკლები ხარისხის მქონე $A(x)$ მრავალწევრების სიმრავლით, სადაც

$$A(x) = A_0 + A_1 x + A_2 x^2 + ... + A_{k-1} x^{k-1},\ A_i \in GF(p),\ k \le n.$$

$\underline{a} = (a_0, a_1, ..., a_{n-1})$ კოდური სიტყვების ელემენტები ტოლია $a_i = A(\alpha^i),$ ანუ

$$C = \left\{ \underline{a} \mid a_i = A(\alpha^i),\ i = 0,1,...., n-1, \deg A(x) < k \right\}.$$

RS კოდის მინიმალური მანძილია $d = n-k+1,$ ხოლო კოდის განზომილება ტოლია k-სი. n რიგის ელემენტისათვის [1]-ში მოცემული განსაზღვრების მიხედვით:

$\alpha^n = 1 \bmod p$.

ამასთან, α-ს ყველა სხვა ხარისხები $\alpha^0, \alpha^1, ..., \alpha^{n-1}$ წარმოადგენენ $GF(p)$ ველის განსხვავებულ ელემენტებს. როდესაც α პრიმიტიული ელემენტია, $n = p-1$. $x^n - 1$ განტოლება მართებულია n რიგის $\alpha \in GF(p)$ ელემენტის ყველა ხარისხისათვის, რაც მოიცავს ყველა $\alpha^i, i = 0,1,...,n-1,$ ელემენტს. მაშასადამე, $x^n - 1$ მრავალწევრს აქვს ზუსტად n განსხვავებული წრფივი მამრავლი $(x - \alpha^i)$, რომელიც წარმოქმნის ამ მრავალწევრს. ამგვარად, მართებულია შემდეგი

თეორემა 10.3. $x^n - 1$ მრავალწევრის წრფივი მამრავლები შეიძლება ჩაიწეროს:

$$x^n - 1 = \prod_{i=0}^{n-1} (x - \alpha^i)$$

სახით, სადაც $\alpha \in GF(p)$-ს n რიგის ელემენტია.

შევნიშნოთ, რომ პრიმიტიული $\alpha \in GF(p)$ ელემენტისთვის $n = p^{-1}$ და $\alpha^{p-1} = 1 = \alpha^0$. შევთანხმდეთ, რომ ქვემოთ ყველგან α ელემენტის ექსპონენტაში მოცემული გამოსახულება გამოითვლება $(p-1)$-ის მოდულით, ხოლო თვითონ ელემენტები და მრავალწევრის კოეფიციენტები p-ს მოდულით. ამასთან, ვინაიდან მართებულია ტოლობა:

$\alpha^n = 1 \rightarrow x^n - 1 = 0$,

$x^n - 1$ მრავალწევრი შეიძლება გამოყენებული იქნას როგორც მოდულით გამყოფი, ნებისმიერი მრავალწევრისათვის. მრავალწევრის კოეფიციენტებისათვის, რომლებიც შეესაბამება x-ის $(n-1)$-ზე მეტ ხარისხებს, $(x^n - 1)$-ის მოდულით ოპე-

რაციას მივყავართ $(i \cdot n + j)$ ხარისხის შემცველი წევრის კოეფიციენტის შეკრებასთან j ხარისხის შემცველი წევრის კოეფიციენტთან x^j, სადაც $i = 1, 2, \ldots$. მაგალითად, $n = 6$ -სთვის:

$$1 + x^3 + x^{12} + x^{29} = x^0 + x^{0 \cdot 6+3} + x^{2 \cdot 6+0} + x^{4 \cdot 6+5}$$
$$= (2 + x^3 + x^5) \bmod (x^6 - 1).$$

ასევე, მრავალწევრის გამრავლებისას x^j ერთწევრზე $(x^n - 1)$ -ის მოდულით ვიღებთ მრავალწევრს, რომელიც წარმოადგენს მამრავლი მრავალწევრის კოეფიციენტების ციკლურ წანაცვლებას i კოორდინატით.

<u>მაგალითი 10.1.</u> ავაგოთ $n = 6$ სიგრძისა და $d = 5$ მინიმალური მანძილის მქონე RS კოდი $GF(7)$ ველზე. თავდაპირველად ჩვენ უნდა ვიპოვოთ ამ ველის პრიმიტიული ელემენტი, ან სხვა სიტყვებით რომ ვთქვათ, ელემენტი რომლის რიგი ტოლია 6-ის. ჩვენ ასევე უნდა განვიხილოთ ყველა ისეთი $A(x)$ მრავალწევრი, რომლის ხარისხიც $k - 1 \le n - d$, ანუ $k \le 2$. თავდაპირველად შევამოწმოთ, არის თუ არა $\alpha = 5$ $GF(7)$ ველის პრიმიტიული ელემენტი:

$$5^1 = 5, \quad 5^2 = 25 = 4, \quad 5^3 = 25 \cdot 5 = 4 \cdot 5 = 20 = 6.$$

ამ მომენტისათვის ჩვენ შეგვიძლია დავასკვნათ, რომ $\alpha = 5$ პრიმიტიული ელემენტია, ვინაიდან [1]-ში მოყვანილი 2.10 თეორემის მიხედვით $GF(p = 7)$ ველის ნებისმიერი ელემენტის რიგი უნდა ყოფდეს $p - 1 = 6$ -ს. ამგვარად, ელემენტების შესაძლო რიგი ტოლია 2, 3 ან 6-ის. ვინაიდან ჩვენ უკვე ვიცით, რომ $\alpha = 5$ ელემენტის რიგი მეტია 3-ზე, ე. ი. მისი რიგი ტოლია 6-ის და, აქედან გამომდინარე, $\alpha = 5$ პრიმი-

ტიული ელემენტია. ახლა განვაგრძოთ $\alpha = 5$-ის ხარისხების გამოთვლა:

$$5^4 = 30 = 2, \quad 5^5 = 10 = 3, \quad 5^6 = 15 = 1.$$

ვუჩვენოთ, რომ ექსპონენტის გამოსახულების $(n-1)$-ის მოდულით გამოთვლას მივყავართ იმავე შედეგებამდე, როგორსაც თვითონ ელემენტების p-ს მოდულით გამოთვლას:

$$5^9 = 5^4 \cdot 5^5 = 2 \cdot 3 = 6 = 5^{9 \bmod 6} = 5^3 = 6$$
$$\neq 5^{9 \bmod 7} = 5^2 = 4 \quad .$$

ახლა ჩვენ უკვე მზად ვართ ავაგოთ RS კოდი. მისი $\underline{a}$ კოდური სიტყვები გამოითვლება $a_i = A(\alpha^i)$ ფორმულით, სადაც $A(x) = A_0 + A_1 x$, ხოლო $A_0, A_1 \in GF(7)$. სულ არსებობს $p^k = 7 \cdot 7 = 49$ განსხვავებული $A(x)$ და, მაშასადამე, 49 კოდური სიტყვა. მაგალითისათვის, $A(x) = 5 + 3x$ საინფორმაციო მრავალწევრის შესაბამისი კოდური სიტყვის გამოთვლა ხდება შემდეგნაირად:

$$a_0 = A(\alpha^0) = A(1) = 5 + 3 = 1 \bmod 7,$$
$$a_1 = A(\alpha^1) = A(5) = 5 + 3 \cdot 5 = 6 \bmod 7,$$
$$a_2 = A(\alpha^2) = A(4) = 5 + 3 \cdot 4 = 3 \bmod 7,$$
$$a_3 = A(\alpha^3) = A(6) = 5 + 3 \cdot 6 = 2 \bmod 7,$$
$$a_4 = A(\alpha^4) = A(2) = 5 + 3 \cdot 2 = 4 \bmod 7,$$
$$a_5 = A(\alpha^5) = A(3) = 5 + 3 \cdot 3 = 0 \bmod 7 .$$

ამგვარად, საბოლოოდ მივიღებთ: $\underline{a} = (1, 6, 3, 2, 4, 0)$.

დავსვათ შემდეგი კითხვა: როგორ შეიძლება $A(x)$ საინფორმაციო მრავალწევრის გამოთვლა კოდური $\underline{a}$ სიტყვის

მიხედვით? $\underline{a}$ ვექტორი წარმოვადგინოთ მრავალწევრის სახით $a(x) = a_0 + a_1 x + a_2 x^2 + ... + a_{n-1} x^{n-1}$. ჩვენი ამოცანაა $a(x)$ მრავალწევრის $A(x)$ მრავალწევრად გარდაქმნისას შესრულებული ასახვის დეტალური წარმოდგენა. აღნიშნულ მრავალწევრებს შორის პირდაპირი და უკუგარდაქმნები შეიძლება აღწერილ იქნას განსხვავებული მათემატიკური აპარატის გამოყენებით. ქვემოთ ჩვენ განვიხილავთ ტექნიკას, რომელიც იყენებს ფურიეს დისკრეტული გარდაქმნის მეთოდს, სადაც $\circ - \bullet$ აღნიშნავს ბიექციას.

განსაზღვრება 10.2. $i, j \in 0,1,...,n-1$ ინდექსებისათვის ფურიეს დისკრეტული გარდაქმნა (DFT) და ფურიეს დისკრეტული უკუგარდაქმნა $(IDFT)$ განისაზღვრება შემდეგნაირად:

$$\left.\begin{array}{l} DFT - a_i = A(\alpha^i) \\ IDFT - A_j = n^{-1} a(\alpha^{-j}) \end{array}\right\} a(x) \circ - \bullet A(x),$$

სადაც $\alpha \in GF(p)$ n რიგის ელემენტია, ხოლო $a(x)$ და $A(x)$ მრავალწევრების კოეფიციენტები მოთავსებულია $GF(p)$ ველში და მათი ხარისხები ნაკლებია ან მეტი $(n-1)$ -ზე.

განვიხილოთ ორი $A(x) = A_0 + A_1 x + ... + A_{n-1} x^{n-1}$ და $B(x) = B_0 + B_1 x + ... + B_{n-1} x^{n-1}$ მრავალწევრი კოეფიციენტებით $GF(p)$-დან. მათი ციკლური ხვევა $A(x) * B(x)$ განისაზღვრება როგორც $C(x) = C_0 + C_1 x + ... + C_{n-1} x^{n-1}$ მრავალწევრი,

სადაც $C_j = \sum_{i=0}^{n-1} A_i B_{j-i}, \quad j = 0,1,...,n-1,$ და ყველა ინდექსი გამოითვლება n-ის მოდულით.

შევნიშნოთ, რომ გაფართოებული $GF(2^m)$ ველებისათვის $n^{-1} = 1$ და გამოთვლები ხორციელდება საბაზისო $GF(2)$ ველში, სადაც $(2^m - 1) \cdot 1 = 1 \bmod 2.$

ორი $a(x)$ და $b(x)$ მრავალწევრის ნამრავლს აღნიშნული გარდაქმნის არეში შეესაბამება ხვევის ოპერაცია.

თეორემა 10.4. ორი $a(x)$ და $b(x)$ მრავალწევრის x^{n-1} მრავალწევრის მოდულით ნამრავლს შეესაბამება ციკლური ხვევა:

$$a(x) \circ\!-\!\bullet\, A(x), \quad b(x) \circ\!-\!\bullet\, B(x);$$

$$c_i = a_i b_i, \quad c(x) \circ\!-\!\bullet\, C(x) = A(x) * B(x)$$
$$= A(x)B(x) \bmod (x^n - 1);$$

$$C_i = A_i B_i, \quad C(x) \bullet\!-\!\circ\, c(x) = \frac{1}{n} a(x)b(x) \bmod (x^n - 1).$$

დამტკიცება: ხვევის განტოლებების დამტკიცება დაფუძნებულია 10.2 განსაზღვრებაზე.

მაგალითისათვის,

$$C_j = \frac{1}{n} c(\alpha^{-j}) = \frac{1}{n} \sum_{i=0}^{n-1} \alpha^{-ij} c_i$$
$$= \frac{1}{n} \sum_{i=0}^{n-1} \alpha^{-ij} a_i b_i = \frac{1}{n} \sum_{i=0}^{n-1} \alpha^{-ij} a_i \left(\sum_{k=0}^{n-1} \alpha^{-ik} B_k \right)$$

$$= \sum_{k=0}^{n-1} B_k \left(\frac{1}{n} \sum_{i=0}^{n-1} \alpha^{-i(j-k)} a_i \right) = \sum_{k=0}^{n-1} B_k A_{j-k}.$$

აქედან გამომდინარე,

$$C(x) = A(x) * B(x) = A(x)B(x) \bmod (x^n - 1). \qquad \square$$

n სიგრძის, k განზომილების და $d = n - k + 1$ მინიმალური მანძილის RS კოდი შეიძლება წარმოვადგინოთ ყველა ისეთი (საინფორმაციო) $i(x)$ მრავალწევრის მეშვეობით, რომელთა ხარისხებიც ნაკლებია k -ზე. კოდური სიტყვები, ანუ შესაბამისი $a(x)$ მრავალწევრები, გამოითვლება $i(x)$ -ის გამრავლებით წარმომქმნელ $g(x)$ მრავალწევრზე:

$$a(x) = i(x)g(x), \quad \deg g(x) = n - k.$$

ამასთან, თითოეული $a(x)$ კოდური სიტყვა უნდა იყოფოდეს $g(x)$ -ზე.

ანალოგიურად, RS კოდი შეიძლება განისაზღვროს როგორც ისეთი $A(x)$ მრავალწევრების სიმრავლე, რომელთა ხარისხებიც ნაკლები ან ტოლია $(k-1)$ -ის. მაშასადამე, კოეფიციენტები $A_i = 0$, სადაც $k \le i \le n-1$. ეს შეესაბამება 10.2 განსაზღვრებით მოცემულ გარდაქმნას:

$$A_i = n^{-1} a(\alpha^{-i}) = 0, \text{ სადაც } k \le i \le n-1.$$

თითოეული კოდური სიტყვის შესაბამისი $a(x)$ მრავალწევრისათვის $x = \alpha^i$ ელემენტები უნდა წარმოადგენდეს მის ფესვებს. ეს შეესაბამება $(x - \alpha^{-i})$ წრფივ მამრავლებს, რომელთა ნამრავლიც გვაძლევს $n-k$ ხარისხის მქონე წარმომქმნელ $g(x)$ მრავალწევრს:

$$g(x)=\prod_{i=k}^{n-1}(x-\alpha^{-i}).$$

ეს ტოლობა მართებულია იმის გამო, რომ 10.4 თეორემის თანახმად $a(x)=i(x)g(x)\circ-\bullet I\cdot G=A,$ სადაც $G_i=0,$ $k\le i\le n-1$ მნიშვნელობებისათვის. RS კოდი ციკლური კოდია და, მაშასადამე, თუ $c(x)\in C$, გვაქვს

$$xc(x)=xi(x)g(x)=i'(x)g(x)\bmod(x^n-1)\in C.$$

<u>მაგალითი 10.2.</u> გამოვთვალოთ წინა 10.1 მაგალითში მოცემული RS კოდის წარმომქმნელი მრავალწევრი. ავირჩიოთ ისევ $\alpha=5$ როგორც $GF(7)$ ველის პრიმიტიული ელემენტი და გავიხსენოთ, რომ $A_i=0,\ i=2,3,4,5.$ ამგვარად, გვაქვს:

$$g(x)=\prod_{i=2}^{5}(x-\alpha^{-i})$$
$$=(x-\alpha^{-2})(x-\alpha^{-3})(x-\alpha^{-4})(x-\alpha^{-5})\Big|\bmod 6$$

(ექსპონენტისათვის);

$$=(x-\alpha^4)(x-\alpha^3)(x-\alpha^2)(x-\alpha)$$
$$=(x^2-(\alpha^4+\alpha^3)x+\alpha^7)(x^2-(\alpha^2+\alpha)x+\alpha^3)$$
$$=(x^2-x+5)(x^2-2x+6)$$
$$=x^4-x^3+5x^2-2x^3+2x^2-10x+6x^2-6x+30\Big|\bmod 7$$

(კოეფიციენტებისათვის);

ანუ $g(x)=x^4+4x^3+6x^2+5x+2.$

10.1 მაგალითში მოცემული კოდური სიტყვა <u>a</u> უნდა ყოფდეს $g(x)$-ს, ე. ი $a(x)/g(x)=i(x)$:

$$(4x^4+2x^3+3x^2+6x+1)/(x^4+4x^3+6x^2+5x+2)=4.$$

ამგვარად, $a(x) = 4g(x)$ და $i(x) = 4 + 0 \cdot x$.

ლუწობის შემმოწმებელი $h(x)$ მრავალწევრი განისაზღვრება შემდეგნაირად:

$a(x)h(x) = 0 \bmod(x^n - 1)$ ყველა $a(x) \in C$ -სთვის.

შესაბამისი გამოსახულება სიხშირულ დომენში, რომელიც დაფუძნებულია 10.2 განსაზღვრებაზე, ჩაიწერება ასე:

$$A_i H_i = 0,\ i = 0,1,...,n-1,\ \ h(x) \circ\!-\!\bullet\, H(x).$$

აქ H_i კოეფიციენტები უნდა იყოს ნულის ტოლი, მაშინ როდესაც შესაბამისი A_i კოეფიციენტები განსხვავდება ნულისგან. ლუწობის შემმოწმებელი მრავალწევრი გამოითვლება წარმომქმნელი მრავალწევრის ანალოგიურად:

$$h(x) = \prod_{i=0}^{k-1} (x - \alpha^{-i}),\ \deg h(x) = k\ .$$

$g(x)$ და $h(x)$ მრავალწევრების ფესვები მიეკუთვნება გადაუკვეთელ სიმრავლეებს. ყველა შესაძლო ფესვი განეკუთვნება ან $g(x)$-ს ან $h(x)$-ს. აქედან გამომდინარე 10.3 თეორემის მიხედვით გვაქვს:

$$g(x)h(x) = \prod_{i=0}^{n-1} (x - \alpha^{-i}) = x^n - 1.$$

<u>მაგალითი 10.3.</u> დავუბრუნდეთ ისევ 10.1 მაგალითში მოყვანილ RS კოდს. შემმოწმებელი $h(x)$ მრავალწევრი ტოლია:

$$h(x) = \prod_{i=0}^{1} (x - \alpha^{-i}) = (x - \alpha^0)(x - \alpha^{-1})$$

$$= (x-1)(x-3) = x^2 + 3x + 3.$$

ამ მრავალწევრის კოდურ $a(x) = 4x^4 + 2x^3 + 3x^2 + 6x + 1$ მრავალწევრზე გამრავლებით ვიღებთ:

$$a(x)h(x) = 0 \bmod (x^6 - 1),$$

ანუ

$$(4x^4 + 2x^3 + 3x^2 + 6x + 1)(x^2 + 3x + 3) =$$

$$\begin{array}{rrrrrr}
 & 4x^6 + 2x^5 + & 3x^4 + 6x^3 + & x^2 & & \\
 & +12x^5 + & 6x^4 + 9x^3 + & 18x^2 + & 3x & \\
 & & +12x^4 + 6x^3 + & 9x^2 + & 18x + 3 & \\
\hline
\bmod 7: \cdot & 14 & 21 & 21 & 28 & 21 \quad \cdot \\
 & = & = & = & = & = \\
 & 0 & 0 & 0 & 0 & 0
\end{array}$$

ჩვენ აღვნიშნავთ $\cdot$ ნიშნით, რომ როდესაც $n = 7$, $GF(7)$ ველისათვის $4x^{6 \bmod (n-1)} + 3x^0$ ანუ $(4+3)x^0 = 7 \bmod 7 = 0.$

RS კოდის გენერირება შეიძლება განხორციელდეს 4 განსხვავებული მეთოდით. ორი მათგანი წარმოადგენს სისტემატურ კოდირებას, სადაც კოდურ სიტყვებში შეგვიძლია განვაცალკეოთ საინფორმაციო და ლუწობის შემმოწმებელი სიმბოლოები. ოთხივე მეთოდით ხდება ერთსა და იმავე კოდის გენერირება, განსხვავდება მხოლოდ საინფორმაციო ვექტორების ასახვის წესი კოდურ სიტყვებში. სხვა სიტყვებით რომ ვთქვათ, თუ ერთი მეთოდის გამოყენებისას მოცემული საინფორმაციო ვექტორი აისახება ერთ კოდურ სიტყვაში, მეორე, განსხვავებული მეთოდის გამოყენებისას იგივე საინფორმაციო ვექტორი აისახება სხვა კოდურ სიტყვაში.

განვიხილოთ RS კოდის გენერირების ოთხივე მეთოდი ცალ-ცალკე:

მეთოდი 1. (არასისტემატური): k საინფორმაციო სიმბოლო წარმოადგენს $A(x) = A_0 + A_1 x + ... + A_{k-1} x^{k-1}$ მრავალწევრის კოეფიციენტებს; კოდური $a(x)$ სიტყვა მიიღება უკუგარდაქმნის მეშვეობით.

მეთოდი 2. (არასისტემატური): k საინფორმაციო სიმბოლო წარმოადგენს $i(x) = i_0 + i_1 x + ... + i_{k-1} x^{k-1}$ მრავალწევრის კოეფიციენტებს. კოდური $a(x)$ სიტყვა ტოლია $a(x) = i(x)g(x)$, სადაც $g(x)$ წარმომქმნელი მრავალწევრია.

მეთოდი 3. (სისტემატური): k საინფორმაციო სიმბოლო შეუცვლელად გადადის კოდურ სიტყვაში და იკავებს $a_{n-k}, a_{n-k+1}, ..., a_{n-1}$ კოორდინატებს; $n-k$ ლუწობის შემმოწმებელი სიმბოლო გამოითვლება შემდეგნაირად:

$$(a_{n-1}x^{n-1} + ... + a_{n-k}x^{n-k}) : g(x) = i(x) + \text{ნაშთი } (x),$$

$$\alpha(x) = a_{n-1}x^{n-1} + ... + a_{n-k}x^{n-k} - \text{ნაშთი } (x).$$

მეთოდი 4. (სისტემატური): k საინფორმაციო სიმბოლო შეუცვლელად გადადის კოდურ სიტყვაში და იკავებს $a_{n-k}, a_{n-k+1}, ..., a_{n-1}$ კოორდინატებს; $n-k$ ლუწობის შემმოწმებელი სიმბოლო გამოითვლება შემდეგნაირად:

$$a_j = \frac{1}{h_0} + \sum_{i=1}^{k} a_{n-i+j} h_j, \quad j = 0,1,...,n-k-1,$$

სადაც $n-i+j$ ინდექსი გამოითვლება n -ის მოდულით, ხოლო $h(x)$ ლუწობის შემმოწმებელი მრავალწევრია.

ქვემოთ, RS კოდების დეკოდირების შესწავლისას ჩვენ ვნახავთ, რომ დეკოდირებისათვის არა აქვს მნიშნელობა იმას, თუ კოდირების რომელი მეთოდი იყო გამოყენებული. თუმ-

ცაღა, ცხადია, რომ კოდირებული სიტყვიდან საინფორმაციო ვექტორის აღსადგენად აუცილებელია გამოყენებული კოდირების მეთოდის ცოდნა. ასეთი პრინციპი მართებულია ყველა ტიპის კოდებისთვის, ამასთან დეკოდირებისას საინფორმაციო სიმბოლოზე შეცდომის ალბათობა დამოკიდებულია საინფორმაციო ვექტორის კოდურ სიტყვებში ასახვის მეთოდზე, ხოლო საინფორმაციო ვექტორზე შეცდომის ალბათობა ერთნაირია ასახვის ყველა მეთოდისათვის.

ჩვენ შეგვიძლია მივიღოთ RS კოდის ეკვივალენტური კოდი, რომელსაც აქვს იგივე მინიმალური მანძილი, როგორიც საწყის კოდს, თუ ავიღებთ n რიგის α ელემენტს და კოდური სიტყვის a_i კოორდინატებს გავამრავლებთ α^{ib} -ზე, $b \in N_0$. ცხადია, რომ $a_i\alpha^{ib} = 0 \cdot a_i = 0$. $A(x) \bullet\!\!-\!\!\circ\, a(x)$ გარდაქმნის მიხედვით: $a_i\alpha^{ib} \circ\!\!-\!\!\bullet\, x^b A(x) \bmod(x^n - 1)$.

x^b მამრავლზე ნამრავლი შეესაბამება $A(x)$ მრავალწევრის ციკლურ წანაცვლებას. $U(x) = x^b A(x) \bmod(x^n - 1)$ ტოლობიდან გვაქვს: $A_i = U_{i+b}$. $A_i = 0$ კოორდინატები ციკლურად არის წანაცვლებული და $(i+b)$ ინდექსი გამოითვლება n -ის მოდულით.

განსაზღვრება 10.3. k განზომილების, n სიგრძის და $d = n - k + 1$ მინიმალური მანძილის მქონე RS კოდი განისაზღვრება შემდეგნაირად: $u(x) \circ\!\!-\!\!\bullet\, U(x)$, სადაც $U(x) = x^b A(x) \bmod(x^n - 1)$ ხოლო $\deg A(x) < k$. ამგვარად,

$$C := \{u(x) \mid u(x) \circ\!\!-\!\!\bullet\, U(x)\}.$$

RS კოდის ყველა კოდური სიტყვის გარდაქმნის შედეგად ვიღებთ $n - k = d - 1$ მიმდევრობით ნულს, რაც უზრუნ-

ველყოფს d -ს ტოლ მინიმალურ მანძილს. α -ს ინდექსები 0-სა და n -სთვის ეკვივალენტურია, ამიტომ მიმდევრობითი კოორდინატები შეიძლება მოცემული იყოს ციკლური სახით: $n-i, n-i+1, ..., n-1, 0, 1, ..., j.$

ავირჩიოთ $GF(p)$ ველის n განსხვავებული არანულოვანი ელემენტი $\alpha_0, \alpha_1, ..., \alpha_{n-1}$ (კერძო შემთხვევაში, ჩვენ შეგვიძლია ავიღოთ n რიგის α ელემენტი და მაშინ სიდიდეები $\alpha^i, i = 0, ..., n-1,$ გვაძლევს $GF(p)$ ველის n განსხვავებულ ელემენტს). ასევე ავირჩიოთ ამავე ველის n განსხვავებული არანულოვანი ელემენტი $\beta_i \in GF(p),\ i = 0, 1, ..., n-1.$

განზოგადებული რიდ-სოლომონის (GRS) კოდი განისაზღვრება შემდეგნაირად:

$$C_{GRS} = \{a(x) \mid a_i = \beta_i A(\alpha_i),\ \beta_i \neq 0,\ i = 0, 1, ..., n-1\},$$

სადაც $A(x) = A_0 + A_1 x + ... + A_{k-1} x^{k-1},\ \ A_i \in GF(p).$

თეორემა 10.5. აღნიშნული განსაზღვრებით მოცემულ GRS კოდს აქვს იგივე პარამეტრები (სიგრძე n, განზომილება k და მინიმალური მანძილი $d = n-k+1$), როგორიც ჩვეულებრივ RS კოდს. თუ ყველა $\beta_i = 1$, ჩვენ მივიღებთ ჩვეულებრივ, 10.1 განსაზღვრებით მოცემულ RS კოდს.

დამტკიცება. 10.1 თეორემის მიხედვით $A(x)$ მრავალწევრს აქვს მაქსიმუმ $k-1$ მიმდევრობითი ნული და ამგვარად, $a(x)$ მრავალწევრს აქვს სულ მცირე $n-k+1 = d$ არანულოვანი კოორდინატი. იმის გამო, რომ მამრავლი $\beta_i \neq 0$, $a(x)$ მრავალწევრის წონა არ იცვლება. □

$GF(p)$ ველზე მოცემული $C(n,k,d)$ RS კოდის ერთი კოორდინატით დაგრძელებით ვიღებთ ახალ კოდს:

$$C_{ext} = \{a_i, i = 0,1,....,n-1, a_n = -\sum_{i=0}^{n-1} a_i = -a(x=1)\},$$

სადაც

$$a(x) \circ - \bullet A(x) = A_0 + A_1 x + ... + A_{k-1} x^{k-1}, A_i \in GF(p).$$

თეორემა 10.6. ერთი კოორდინატით დაგრძელებულ RS კოდს აქვს სიგრძე $n+1$, განზომილება k და მინიმალური მანძილი $d = n-k+2$; ამგვარად,

$$C_{ext}(n+1,k,d+1) = C_{ext}(p,k,n-k+2).$$

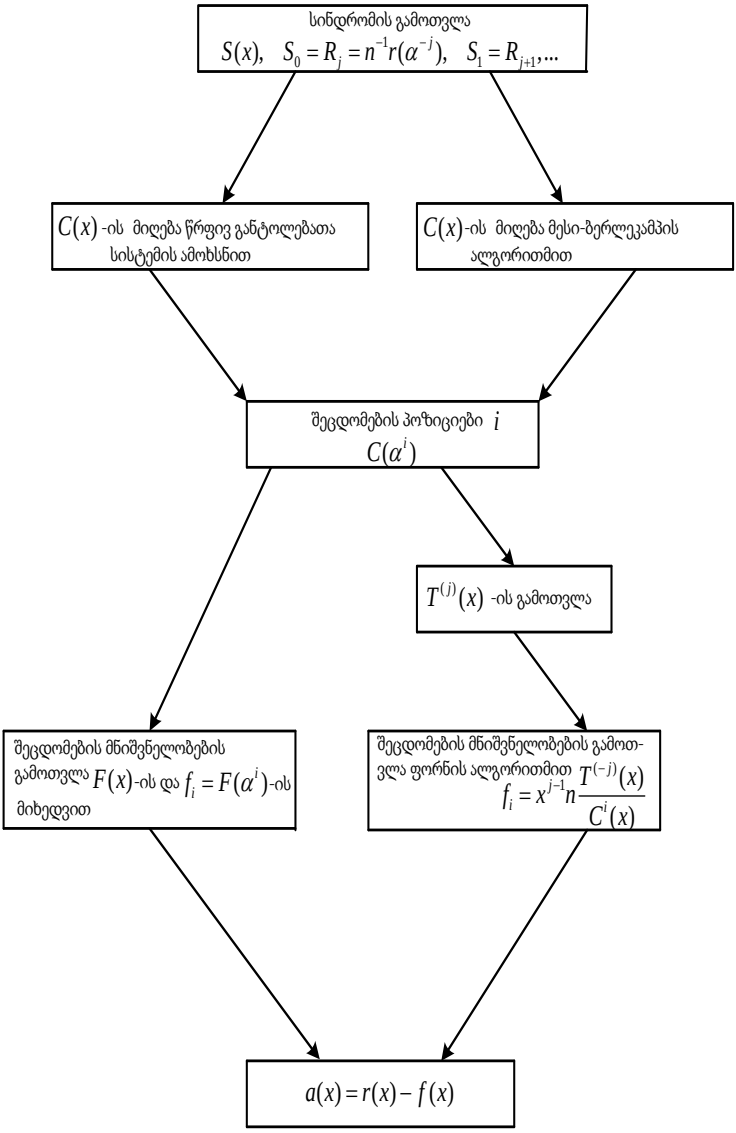
დამტკიცება. ავიღოთ მინიმალური მანძილის მქონე $a(x) \in C(n,k,d)$ კოდური სიტყვა და α^i ფესვებისაგან, $i \in 1,2,...,d-1$, აგებული წარმომქმნელი მრავალწევრი. განვიხილოთ ორი შემთხვევა, $a_n = 1$ და $a_n = 0$. როდესაც $a_n = 1$ კოდური სიტყვის წონა და, მაშასადამე, კოდის წრფივობის გამო, მისი მინიმალური მანძილიც, იზრდება ერთით. აქედან გამომდინარე, კოდის მინიმალური მანძილი ტოლია $(d+1)$-ის. ახლა განვიხილოთ $a_n = 0$ შემთხვევა. განსაზღვრების მიხედვით $a(x) = i(x)g(x)$. იმისათვის, რომ შესრულდეს $a(x=1) = 0$ პირობა, ან $i(x=1)$ უნდა უდრიდეს 0-ს, ან $g(x=1)$ უნდა უდრიდეს 0-ს, ან ორივე ერთად. $g(x=1)$ არ შეიძლება უდრიდეს 0-ს ჩვენ მიერ დაწესებული შეზღუდვების თანახმად. თუ $i(x=1)$ უდრის 0-ს, მაშინ $i(x) = (x-1)\tilde{i}(x)$ და $a(x) = i(x)g(x) = \tilde{i}(x)(x-1)g(x)$.

ამგვარად, ჩვენ შეგვიძლია თავიდან განვსაზღვროთ დაგრძელებული კოდის წარმომქმნელი მრავალწევრი, როგორც $\tilde{g}(x) = (x-1)g(x)$. ახალი წარმომქმნელი $\tilde{g}(x)$ მრავალწევრის მამრავლები შეიცავს α^0 ფესვების მამრავლს, რაც 10.2 თეორემის თანახმად ერთით ზრდის კოდის მინიმალურ მანძილს. ამასთან, ცხადია, რომ კოდური $a(x)$ მრავალწევრი იყოფა ახალ წარმომქმნელ $\tilde{g}(x)$ მრავალწევრზე და, ამიტომ, ის წარმოადგენს კოდურ სიტყვას, რომელიც ეკუთვნის $C_{ext}(n+1, k, d+1)$ კოდს. □

დაგრძელებული RS კოდი არ არის ციკლური კოდი. DFT-ს გამოყენებით ჩვენ შეგვიძლია a_n კოორდინატის განსაზღვრა შემდეგნაირად: $-nA(x=0)$ და, აქედან გამომდინარე, $A_0 = n^{-1}a(x=1)$. შესაძლებელია RS კოდების ორი კოორდინატითაც დაგრძელება, რის შედეგადაც ვიღებთ $C_{ext}(n+2, k, d+2)$ კოდს, რომელიც ისევ ციკლური კოდია.

11. რიდ-სოლომონის კოდების ალგებრული დეკოდირება

მე-4 ნახაზზე მოცემულია RS კოდების ალგებრული დეკოდირების ალგორითმის ბლოკ-სქემა.



ნახ. 4. ალგებრული დეკოდირების ალგორითმი

ჩვენ დეტალურად შევისწავლით დეკოდირებისას შესრულებულ თითოეულ პროცედურას.

განვიხილოთ RS კოდი, რომლის კოდური $a(x)$ სიტყვები მოცემულია 10.2 განსაზღვრების თანახმად, როგორც $a(x) \circ\!-\!\bullet A(x),\ A_0 = A_1 = A_2 = ... = A_{d-2} = 0$.

$A(x)$ მრავალწევრის ზუსტად $d-1$ მიმდევრობითი კოეფიციენტი ტოლია ნულის და ამიტომ RS კოდის მანძილია d და მას შეუძლია გაასწოროს $t \le \left\lfloor \frac{d-1}{2} \right\rfloor$ შეცდომა.

დავუშვათ, რომ ჩვენ მივიღეთ $r(x) = a(x) + f(x)$ ვექტორი, სადაც f_i კოეფიციენტები მდებარეობენ $GF(q)$ ველში. შეცდომის $f(x)$ მრავალწევრის კოეფიციენტები $f_i \neq 0$, მიღებული ვექტორის ყველა იმ პოზიციისთვის, სადაც ადგილი აქვს შეცდომას. ასევე დავუშვათ, რომ გადაცემისას ადგილი ჰქონდა $\left\lfloor \frac{d-1}{2} \right\rfloor$-ის ტოლ ან ნაკლები რაოდენობის შეცდომებს, ე. ი. $wt(\underline{f}) \le \left\lfloor \frac{d-1}{2} \right\rfloor$. შეცდომების პოზიციები აღვნიშნოთ ე. წ. მხარდამჭერი (*support, sup*) სიმრავლით, $sup(\underline{f}) := \{i \mid f_i \neq 0\}$. იმის გასარკვევად, არის თუ არა $r(x)$ კოდური სიტყვა, $r(x)$ მრავალწევრი გარდაიქმნება DFT-ს საშუალებით და მიღებული $R(x)$ მრავალწევრისთვის მოწმდება $R_0 = ... = R_{d-2} = 0$ ტოლობების მართებულობა. ეს, ცხადია, თუ გავითვალისწინებთ, რომ

$a(x)+f(x)=r(x)\circ-\bullet R(x)=A(x)+F(x)$.

თუ $f(x)=0$, მაშინ $R(x)=A(x)$ და $R_0,...,R_{d-2}$ კოეფიციენტები ნულის ტოლია. მეორე მხრივ, თუ $f(x)\neq 0$, ჩვენ შეგვიძლია განვსაზღვროთ $S(x)=S_0+S_1x+....+S_{d-2}x^{d-2}$ სინდრომის მრავალწევრი, სადაც გვაქვს $R_i=F_i=S_i$, $i=0,1,...,d-2$, რაც იმას ნიშნავს, რომ მე-4 ნახაზზე $j=0$. $S(x)$ სინდრომი დამოკიდებულია მხოლოდ გარდაქმნილ შეცდომის $F(x)$ მრავალწევრზე, ვინაიდან განსაზღვრების მიხედვით: $A_0=A_1=...=A_{d-2}=0$.

მოცემული არხისათვის კოდის არჩევისას ჩვენ უნდა ვეცადოთ, რომ არხში შეცდომათა რიცხვის მოსალოდნელი მნიშვნელობა იყოს კოდის მინიმალური მანძილის ნახევარზე ნაკლები. ამ შემთხვევაში დეკოდირების პრობლემა შეიძლება ფორმულირებული იქნას, როგორც ყველა შესაძლო შეცდომის $f(x)$ მრავალწევრთა შორის, რომელთაც შეეძლოთ $r(x)$-ის გენერირება, ისეთი მათგანის ამორჩევის ამოცანა, რომელსაც აქვს ყველაზე უფრო ნაკლები არანულოვანი კოეფიციენტი. ეს ეკვივალენტურია შეცდომების ისეთი $\underline{f}$ ვექტორის მოძებნის ამოცანისა, რომელსაც აქვს მინიმალური წონა. სამწუხაროდ, „უმცირესი წონის" თვისება არ შეიძლება გამოყენებული იქნას ალგებრულად, ამიტომაც ჩვენ ვიხილავთ შესაბამის „უმცირესი ხარისხის" თვისებას.

შეცდომების ლოკატორი მრავალწევრის განსაზღვრებისას ვითვალისწინებთ, რომ $c_i=0\Leftarrow f_i\neq 0$, ვინაიდან $c_i f_i=0$. $c(x)$ მრავალწევრის კოეფიციენტები ნულის ტოლია

შეცდომების ლოკაციის ადგილებში და ისინი ნებისმიერია არა-შეცდომების (სწორად მიღების) ლოკაციის ადგილებში. ეს განსაზღვრავს მრავალწევრების ფართო კლასს. მრავალწევრთა ამ კლასიდან ჩვენ ვირჩევთ ისეთებს, რომელთათვისაც $C(x)$-ის ხარისხი $\left(c(x) \circ\!-\!\bullet\, C(x)\right)$ ტოლია შეცდომათა რიცხვისა. ეს გულისხმობს, რომ 10.2 განსაზღვრების მიხედვით ჩვენ შეგვიძლია ავსახოთ $c(x)$ მრავალწევრი $C(x)$ მრავალწევრში შემდეგნაირად:

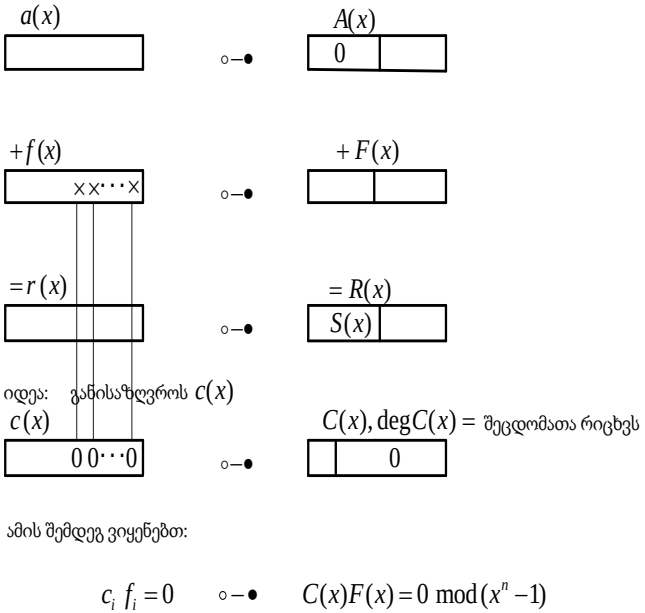
$$C(x) := \prod_{i \in sup(\underline{f})} (x - a^i). \qquad (11.1)$$

$C(x)$-ის ხარისხი ტოლია $c(x)$-ის ნულოვანი კოეფიციენტების რიცხვის, ან რაც იგივეა, შეცდომის $f(x)$ ვექტორში შეცდომების რიცხვის. თუ გამოვიყენებთ 10.4 თეორემას, მივიღებთ: $c_i f_i = 0 \circ\!-\!\bullet\, C(x)F(x) = 0 \bmod(x^n - 1)$.

როდესაც არხის მიერ შემოტანილი e შეცდომების რიცხვი $e \le \left\lfloor \frac{d-1}{2} \right\rfloor$, $C(x)$ მრავალწევრს (11.1) ფორმულის შესაბამისად აქვს სახე: $C(x) = C_0 + C_1 x^1 + ... + C_e x^e$. აქედან გამომდინარე, აუცილებელია განისაზღვროს $C(x)$-ის $e+1$ კოეფიციენტი $C_0, C_1, ..., C_e$. ვინაიდან ჩვენ გვაინტერესებს მრავალწევრის მხოლოდ e „ნული", $C(x)$-ის ნებისმიერი ერთი კოეფიციენტი ჩვენ შეგვიძლია ავირჩიოთ თავისუფალი არჩევის წესით. (11.1) ფორმულის მიხედვით C_e ტოლია 1-ის. მეორე მხრივ, ჩვენ შეგვიძლია ავირჩიოთ $C_0 = 1$ და ასეთ შემთხვევაში მივიღოთ $C(x)$ მრავალწევრის სხვა სახით წარმოდგენა:

$$C(x) := \prod_{i \in \sup(\underline{f})} (1 - a^{-i} x).$$

შევნიშნოთ, რომ $C(x)$-ის ნებისმიერი ნორმირება არ ცვლის მისი ფესვების მნიშვნელობებს. ალგებრული დეკოდირების სტრატეგია სქემატურად მოცემულია მე-5 ნახაზზე, სადაც $a(x)$ - გადაცემული კოდური სიტყვაა, $f(x)$ - შეცდომის სიტყვაა (× განსაზღვრავს შეცდომის ლოკაციას), $r(x) = a(x) + f(x)$ - მიღებული სიტყვაა.



ამის შემდეგ ვიყენებთ:

$$c_i \; f_i = 0 \qquad \circ\!-\!\bullet \qquad C(x)F(x) = 0 \bmod (x^n - 1)$$

ნახ. 5. შეცდომების გასწორების კონცეფცია

უმცირესი შესაძლო ხარისხის მქონე შეცდომების ლოკატორი $C(x)$ მრავალწევრი გამოითვლება $S(x)$ სინდრომის მიხედვით.

ახლა გავარკვიოთ, თუ როგორ შეიძლება შეცდომების ლოკატორი მრავალწევრის კოეფიციენტების გამოთვლა. თავდაპირველად დავუშვათ, რომ არხში ადგილი ჰქონდა ზუსტად $t = \left\lfloor \frac{d-1}{2} \right\rfloor$ შეცდომას. მრავალწევრების ნამრავლი

$$C(x)F(x) = 0 \bmod(x^n - 1) \tag{11.2}$$

შეიძლება ჩაიწეროს წრფივ განტოლებათა სისტემის სახით, სადაც ვითვალისწინებთ იმ ფაქტს, რომ $S(x)$ სინდრომი ცნობილია და განისაზღვრება $F_i = S_i,\ \ i = 0,1,..,d-2,$ ფორმულით. შესაბამისად, გვაქვს

$$\begin{aligned}
&C_0S_0 + C_1F_{n-1} + C_2F_{n-2} + ... + C_tF_{n-t} = 0\\
&C_0S_1 + C_1S_0 + C_2F_{n-1} + ... + C_tF_{n-t+1} = 0\\
&\qquad\vdots\\
&C_0S_{t-1} + C_1S_{t-2} + C_2S_{t-3} + ... + C_tF_{n-1} = 0\\
*\,&C_0S_t + C_1S_{t-1} + C_2S_{t-2} + ... + C_tS_0 = 0\\
*\,&C_0S_{t+1} + C_1S_t + C_2S_{t-1} + ... + C_tS_1 = 0\\
&\qquad\vdots\\
*\,&C_0S_{2t-1} + C_1S_{2t-2} + C_2S_{2t-3} + ... + C_tS_{t-1} = 0\\
&C_0F_{2t} + C_1S_{2t-1} + C_2S_{2t-2} + ... + C_tS_t = 0\\
&\qquad\vdots\\
&C_0F_{n-1} + C_1F_{n-2} + C_2F_{n-3} + ... + C_tF_{n-t-1} = 0.
\end{aligned} \tag{11.3}$$

t განტოლება, რომელსაც აქვს * ნიშანი შეიცავს მხოლოდ სინდრომის მრავალწევრის ცნობილ კოეფიციენტებს და შეცდომების ლოკატორი მრავალწევრის უცნობ კოეფი-

ციენტებს (ერთი კოეფიციენტი შეიძლება ნებისმიერად იქნას არჩეული).

ვინაიდან ჩვენ გვაქვს t უცნობიანი t წრფივ განტოლებათა სისტემა, მისი ამოხსნა თეორიულად შესაძლებელია. $C(x)F(x) = 0 \bmod(x^n - 1)$ გამოსახულებიდან გამომდინარე, ჩვენ გამოვიყვანთ „გასაღებ“ განტოლებას შემდეგნაირად. განვიხილოთ $C(x)S(x)$ ნამრავლი:

$$
\begin{aligned}
&0: C_0S_0\\
&1: C_0S_1 + C_1S_0\\
&\quad\vdots\\
&t-1: C_0S_{t-1} + C_1S_{t-2} + \ldots + C_{t-1}S_0\\
&t: C_0S_t + C_1S_{t-1} + C_2S_{t-2} + \ldots + C_tS_0\\
&t+1: C_0S_{t+1} + C_1S_t + C_2S_{t-1} + \ldots + C_tS_1\\
&\quad\vdots\\
&2t-1: C_0S_{2t-1} + C_1S_{2t-2} + C_2S_{2t-3} + \ldots + C_tS_{t-1}\\
&2t: C_1S_{2t-1} + C_2S_{2t-2} + \ldots + C_tS_t\\
&\quad\vdots\\
&3t-1: C_tS_{2t-1}.
\end{aligned}
\qquad (11.4)
$$

$C(x)S(x)$ ნამრავლით მიღებული $0,1,\ldots,t-1$ კოეფიციენტები (11.3) განტოლებათა სისტემის თანახმად არ გამოდგება $C(x)$ მრავალწევრის განსასაზღვრად. ეს კოეფიციენტები არ უნდა იყოს ნულის ტოლი, თუმცა ზოგიერთი მათგანი შეიძლება იყოს ნულოვანიც. $C(x)S(x)$-ის ნამრავლით მიღებულ $0,1,\ldots,t-1$ კოეფიციენტები აღვნიშნოთ შესაბამისად $T_0, -T_1, \ldots, -T_{t-1}$ სიდიდეებით, რაც ეკვივალენ-

ტურია ახალი $-T(x) = -T_0 - T_1 x - ... - T_{t-1} x^{t-1}$ მრავალწევრის აგებისა. $T(x)$ მრავალწევრს ეწოდება შეცდომების შემფასებელი მრავალწევრი და მარტო ის არ გამოდგება შეცდომების ლოკატორი მრავალწევრის $C(x)$ ამოსახსნელად.

$C(x)S(x)$ ნამრავლით მიღებული $2t, 2t+1, .., 3t-1$ კოეფიციენტები, მიღებული (11.3) განტოლებათა სისტემიდან, ასევე არ არის მნიშვნელოვანი $C(x)$ მრავალწევრის განსასაზღვრად. გამოვთვლით რა $C(x)S(x)$ ნამრავლს x^{2t} -ს მოდულით, ეს კოეფიციენტები ლიკვიდირებული იქნებიან (შევნიშნოთ, რომ სიდიდე $x^{n+1} \bmod (x^n - 1)$ გვაძლევს x -ს, მაგრამ $\bmod x^{2t}$ -ის მიხედვით გამოთვლისას ვიღებთ $x^{2t+i} = 0,\ i = 0,1,...)$. ასეთი მსჯელობით მივდივართ „გასაღებ“ განტოლებებამდე:

$$C(x)S(x) = -T(x) \bmod x^{2t}. \qquad (11.5)$$

როგორც უკვე აღვნიშნეთ, (11.3) ფორმულით მოცემულ წრფივ განტოლებათა სისტემის აუცილებელი ნაწილი (მოცემული * ნიშნით) მონაწილეობას იღებს „გასაღებ“ განტოლებებში, სახელდობრ,

$$0 = \sum_{i=0}^{t} C_i S_{j-i}, \qquad j = t, t+1, .., 2t-1.$$

ორივე ასეთი წარმოდგენა შეიძლება გამოყენებული იქნას $C(x)$ -ის დასადგენად.

ახლა განვიხილოთ შემთხვევა, როდესაც ადგილი აქვს e შეცდომას, $e < t = \left\lfloor \frac{d-1}{2} \right\rfloor$. თუ $e = t-1$ მაშინ $C_t = 0$ და (11.3) ფორმულით მოცემულ წრფივ განტოლებათა სის-

ტემიდან გამომდინარე, ჩვენ გვჭირდება $C(x)$-ის მხოლოდ $t-1$ კოეფიციენტის განსაზღვრა (ერთი კოეფიციენტი შეიძლება ნებისმიერად იქნას არჩეული). ამ შემთხვევაშიც არსებობს (11.3) ფორმულაში * ნიშნით აღნიშნული t განტოლება, რომლებიც შეიცავს უკვე გამოთვლილი სინდრომის მრავალწევრის კოეფიციენტებს და $C(x)$-ის უცნობ კოეფიციენტებს. ახლა ამ განტოლებას შეგვიძლია დავუმატოთ

$$C_0S_{t-1}+C_1S_{t-2}+C_2S_{t-3}+\ldots+C_{t-1}S_0=0$$

განტოლებაც, რომელიც ასევე შედგება სინდრომის მრავალწევრის ცნობილი კოეფიციენტებიდან. ასეთ შემთხვევაში წრფივ განტოლებათა სისტემაში განტოლებათა რიცხვი აღემატება უცნობ კოეფიციენტთა რიცხვს. სულ გვაქვს $t-1$ უცნობი სიდიდე და $t+1$ განტოლება. ასევე (11.4) ფორმულით მოცემულ გამოსახულებაში ჩვენ გვაქვს დამატებითი განტოლება, რომელიც შეესაბამება $t-1$ კოეფიციენტს. ეს ნიშნავს, რომ $-T_{t-1}=0$.

საზოგადოდ, არხში $e\le t=\left\lfloor\frac{d-1}{2}\right\rfloor$ რაოდენობის შეცდომების შემთხვევაში $C(x)$ მრავალწევრის განსასაზღვრად, ჩვენ უნდა განვსაზღვროთ e უცნობი კოეფიციენტი t განტოლების მიხედვით.

აღვნიშნოთ, რომ განსაზღვრული კოეფიციენტები უნდა აკმაყოფილებდნენ ყველა იმ განტოლებას, რომელიც შედგება მხოლოდ C_i და S_i სიდიდეებისაგან. ისევ ჩვენ გვაქვს $2t-e$ ასეთი განტოლება. წრფივ განტოლებათა სისტემის შესაბამისი ნაწილი ჩაიწერება შემდეგნაირად:

$$\begin{pmatrix} S_e & \dots & S_1 & S_0 \\ S_{e+1} & \dots & S_2 & S_1 \\ \vdots & & & \vdots \\ \vdots & & & \vdots \\ S_{2t-1} & \dots & S_{2t-e} & S_{2t-e-1} \end{pmatrix} \cdot \begin{pmatrix} C_0 \\ C_1 \\ \vdots \\ C_{e-1} \\ 1 \end{pmatrix} = 0 . \qquad (11.6)$$

როდესაც სრულდება უტოლობა $e < t$, „გასაღებ“ განტოლებათა გამოსახულებაში ასევე ადგილი უნდა ჰქონდეს უტოლობას: $\deg T(x) < \deg C(x)$.

„გასაღებ“ განტოლებათა ამოხსნას მივყავართ იმავე შედეგებამდე, როგორსაც (11.3) წრფივ განტოლებათა სისტემის ამოხსნას.

განსაზღვრება 11.1. შეცდომების ლოკატორი $C(x)$ მრავალწევრის გამოსათვლელად გამოყენებული „გასაღები“ განტოლება განისაზღვრება $C(x)S(x) = -T(x) \bmod x^{2t}$ გამოსახულებით, სადაც $\deg T(x) < \deg C(x)$ და $C(x)$-ის ხარისხი ტოლია არხში წარმოქმნილ შეცდომათა რიცხვისა.

ახლა ვუჩვენოთ, თუ როგორ შეიძლება გამოითვალოს $C(x)$ მრავალწევრი წრფივ განტოლებათა სისტემის მიხედვით. ცხადია, რომ მიღებულ კოდურ სიტყვაში არსებულ შეცდომათა რიცხვი უცნობია და მისი დადგენა შესაძლებელია ვარაუდისა და ამ ვარაუდის შემოწმების საფუძველზე. ანუ თუ არხიდან მიღებული სიტყვა არ არის კოდური სიტყვა, ჩვენ თავდაპირველად ვვარაუდობთ, რომ ადგილი ჰქონდა $e = 1$ შეცდომას და ვამოწმებთ, აკმაყოფილებს თუ არა ეს გადაწყვეტილება (11.6) ფორმულით მოცემულ ყველა განტოლებას. საწინააღმდეგო შემთხვევაში ჩვენ ვვარაუდობთ, რომ

ადგილი ჰქონდა $e=2$ შეცდომას და ა. შ. კოდურ სიტყვაში არსებულ შეცდომათა რიცხვის დადგენა აგრეთვე შესაძლებელია (11.6) ფორმულით მოცემული სინდრომის კოეფიციენტების მატრიცის რანგის მიხედვით. ამ მატრიცის რანგი შეესაბამება კოდურ სიტყვაში არსებულ შეცდომათა რიცხვს.

სანამ კონკრეტული მაგალითის განხილვაზე გადავიდოდეთ, ჩავთვალოთ, რომ RS კოდი წარმოდგენილია ისე, რომ მისი ყველა კოდური $a(x)$ სიტყვისათვის სრულდება $A_k = A_{k+1} = ... = A_{n-1} = 0$ პირობა. $C(x)F(x) \bmod (x^n - 1)$ გამოსახულების გამრავლება x-ის ნებისმიერ ხარისხზე იწვევს (11.3) განტოლების ხელახალ ციკლურ მოწესრიგებას. თუ

$$S_0 = R_i = F_i,\ S_1 = R_{i+1} = F_{i+1}, ..., S_{d-2} = R_{d-2+i} = F_{d-2+i}$$

სიდიდეებს გამოვიყენებთ, როგორც $S(x)$ მრავალწევრის კოეფიციენტებს, შევნიშნავთ, რომ (11.4) განტოლებები სამართლიანია i-ს ნებისმიერი მნიშვნელობისათვის, სადაც $i \in \{0,1,....,n-1\}$.

„გასაღები“ განტოლების განსაზღვრების თანახმად, მის ამოსახსნელად აუცილებელია მხოლოდ სინდრომის ცოდნა. ამასთან, $S(x)$ სინდრომის ნებისმიერი ადგილმდებარეობისთვის, ჩვენ ვიღებთ $C(x)$-ის ერთსა და იმავე ამონახსნს.

<u>მაგალითი 11.1.</u> დავუშვათ, რომ გადაიცემა $\underline{a} = (1,6,3,2,4,0)$ კოდური სიტყვა (იხ. მაგალითი 10.1). ჩვენ ვირჩევთ $e = \left\lfloor \frac{d-1}{2} \right\rfloor = 2$. დავუშვათ, რომ გადაცემისას ადგილი აქვს $f(x) = 5x^4 + 3x$ შეცდომას. შესაბამისად, მიღებული სიტყვა ტოლია $r(x) = a(x) + f(x) = 2x^4 + 2x^3 + 3x^2 + 2x + 1$

მრავალწევრის. მიმღებ მხარეს ჩვენ ვიცით მხოლოდ $r(x)$ და გამოყენებული RS კოდი. მიღებული ვექტორისათვის გამოვთვლით $S(x)$ სინდრომს. ჩავთვალოთ, რომ $\alpha = 5$ პრიმიტიული ელემენტია. მაშინ ამ გამოთვლას აქვს შემდეგი სახე:

$$\begin{aligned}
S_0 = R_2 &= n^{-1} r(a^{-2}) = 6r(a^4) \\
&= 6(2a^{16} + 2a^{12} + 3a^8 + 2a^4 + 1) \\
&= 6(2a^4 + 2a^0 + 3a^2 + 2a^4 + 1) \\
&= 6(2\cdot 2 + 2\cdot 1 + 3\cdot 4 + 2\cdot 2 + 1) \\
&= 6(4 + 2 + 12 + 4 + 1) = 6\cdot 23 = 5 \bmod 7, \\
S_1 = R_3 &= 6r(a^3) = 6(2a^{12} + 2a^9 + 3a^6 + 2a^3 + 1) \\
&= 6(2a^0 + 2a^3 + 3a^0 + 2a^3 + 1) \\
&= 6(2 + 12 + 3 + 12 + 1) = 5 \bmod 7, \\
S_2 &= R_4 = 3, \\
S_3 &= R_5 = 3, \\
&\Rightarrow S(x) = 5 + 5x + 3x^2 + 3x^3.
\end{aligned}$$

თავდაპირველად დავუშვათ, რომ ადგილი ჰქონდა $e = 1$ შეცდომას: $C(x) = C_0 + x$. წრფივ განტოლებათა სისტემა, რომელიც უნდა დაკმაყოფილდეს, გამოითვლება (11.6) ფორმულის გამოყენებით:

$$S_1 C_0 + S_0 = 0 = 5C_0 + 5 = 0 \Rightarrow C_0 = 6 \text{ და } 5\cdot 6 + 5 = 0 \bmod 7.$$

ჩვენ ვიღებთ $C(x) = 6 + x$ და ვამოწმებთ თუ ასევე კმაყოფილდება $S_2 6 + S_1 = 0$ და $S_3 6 + S_2 = 0$ განტოლებები, მაგრამ ჩვენ ვხედავთ, რომ $S_2 6 + S_1 = 3\cdot 6 + 5 = 2 \bmod 7$. აქედან

გამომდინარე, უნდა დავუშვათ, რომ ადგილი ჰქონდა $e = 2$ შეცდომას: $C(x) = C_0 + C_1 x + x^2$. შესაბამისად, გვექნება:

$$S_2 C_0 + S_1 C_1 + S_0 = 0$$
$$S_3 C_0 + S_2 C_1 + S_1 = 0$$
$$\left.\begin{array}{l} 3C_0 + 5C_1 + 5 = 0 \\ 3C_0 + 3C_1 + 5 = 0 \end{array}\right| \begin{array}{l} 2C_1 = 0 \\ C_1 = 0 \\ C_0 = 3 \end{array}$$

საბოლოოდ ვიღებთ, $C(x) = 3 + x^2$.

$C(x)$ -ის ორი ფესვი განსაზღვრავს ორი შეცდომის ლოკაციის ადგილებს. მათ საპოვნელად ვისარგებლოთ $C(x)$ ველის ყველა შესაძლო ელემენტით:

$1:\ 3+1=4;\quad 2:\ 3+4=0 \bmod 7 \Rightarrow$ პირველი ფესვია: $2=\alpha^4$;

$3:\ 3+9=5 \bmod 7;\quad 4:\ 3+16=5 \bmod 7;$

$5:\ 3+25=0 \bmod 7 \Rightarrow$ მეორე ფესვია: $5=\alpha^1$.

შესაბამისად, $C(x) = (x-2)(x-5) = (x-\alpha^4)(x-\alpha^1)$. ამგვარად, შეცდომითი სიმბოლოების კოორდინატებია 1 და 4, ანუ $f(x) = f_4 x^4 + f_1 x$ (შეცდომის მნიშვნელობების გამოთვლის პროცედურა განხილული იქნება ოდნავ მოგვიანებით). მოყვანილ მაგალითში $C(x)$ მრავალწევრის ფესვები ნაპოვნი იყო მასში $GF(7)$ ველის ყველა ელემენტის ჩასმით. ეს პროცედურა ცნობილია ჩენის ალგორითმის სახელწოდებით. აღვნიშნოთ, რომ თუ ფესვთა რიცხვი ნაკლებია $C(x)$ მრავალწევრის ხარისხზე, მაშინ ადგილი აქვს დეკოდირებაზე უარის თქმას. ეს ნიშნავს, რომ როდესაც არხში არსებული შეცდომების რიცხვი აღემატება კოდის შესაძლებლობე-

ბიდან გამომდინარე გარანტირებულად გასწორებულ შეცდომათა მაქსიმალურ რიცხვს, მაშინ შეიძლება ადგილი ჰქონდეს დეკოდირებაზე უარის თქმას.

ახლა შევისწავლოთ $C(x)$ მრავალწევრის განმსაზღვრელი კიდევ ერთი (მესი-ბერლეკამპის) ალგორითმი, რომლის მიხედვით ვპოულობთ ისეთი უმცირესი ხარისხის მქონე მრავალწევრს, რომელიც აკმაყოფილებს (11.5) განტოლებას. აღნიშნული ალგორითმი, შეიძლება ინტერპრეტირებული იყოს როგორც უმცირესი სიგრძის მქონე უკუკავშირიანი წანაცვლების რეგისტრის მოძებნის ამოცანა, რომლის უკუკავშირის კოეფიციენტებია C_i და რომლის მიხედვითაც შესაძლებელია სინდრომის ყველა კოეფიციენტის გენერირება. ალგორითმი იტერაციულია და მის საწყის ბიჯზე განიხილება შეცდომების ლოკატორი $C(x)$ მრავალწევრი ხარისხით 1. $C(x)$-ის ხარისხი იზრდება მაქსიმუმ ერთით თითოეული იტერაციის დროს. ყოველი მომდევნო $C_i(x)$ წონა გამოითვლება $C_{i-1}(x)$ მრავალწევრისაგან. ალგორითმის ბლოკ-სქემა მოყვანილია მე-6 ნახაზზე.

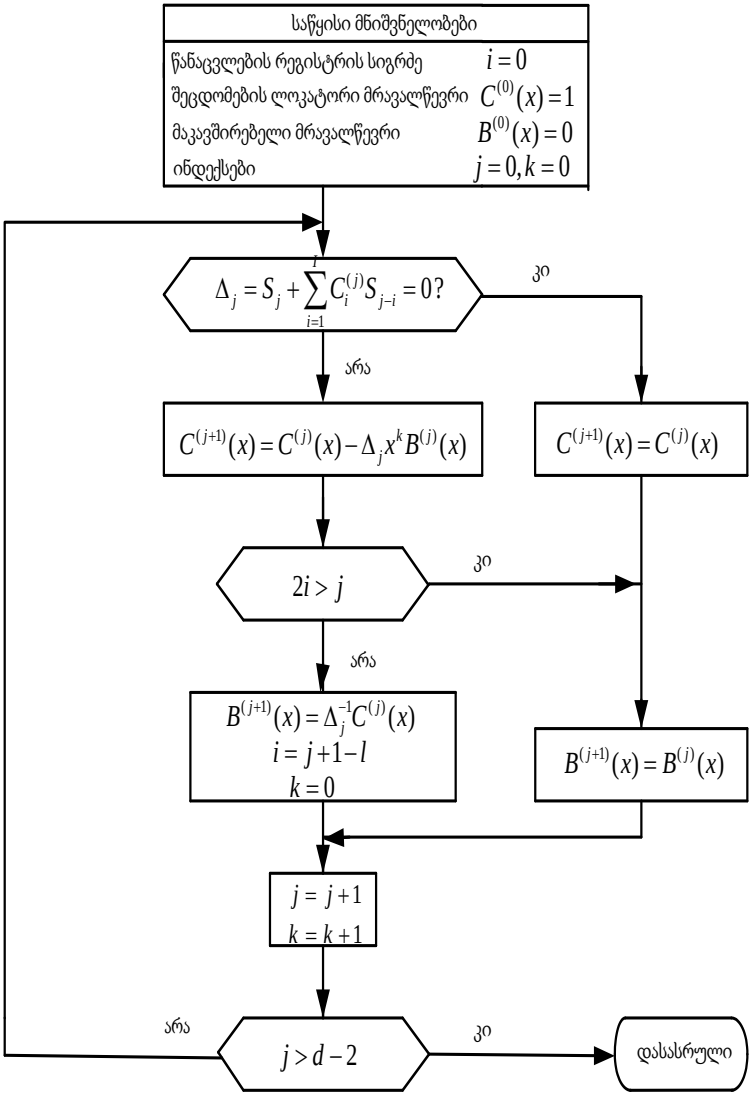
<u>მაგალითი 11.2.</u> ვიპოვოთ შეცდომების ლოკატორი $C(x)$ მრავალწევრი $S(x) = 5 + 5x + 3x^2 + 3x^3$ სინდრომის მიღების შემთხვევაში (იხ. მაგალითი 11.1). მესი-ბერლეკამპის ალგორითმით შეცდომების ლოკატორი მრავალწევრის გამოთვლის მთელი პროცედურა მოცემულია მე-2 ცხრილში. შეცდომების კოორდინატებია 1 და 4. იმის გამო, რომ მესი-ბერლეკამპის ალგორითმის გადაწყვეტილება აუცილებლად შეიცავს $C_0 = 1$ წევრს, ამ ალგორითმით შეიძლება მივიღოთ

წინა მაგალითში განხილული წრფივ განტოლებათა სისტემის ამოხსნით მიღებული გადაწყვეტილების ეკვივალენტური შედეგი წრფივი მამრავლის სიზუსტით. 11.1 მაგალითში ჩვენ მივიღეთ $C(x) = 3 + x^2$, ხოლო 11.2 მაგალითში კი მესი-ბერლეკამპის მეთოდით $C_{MB}(x) = 1 + 5x^2$. $C(x) = 3C_{MB}(x)$ და აღნიშნული მამრავლი არ ცვლის ფესვების მნიშვნელობებს. მესი-ბერლეკამპის ალგორითმი წარმოადგენს შეცდომების ლოკატორი მრავალწევრის განსაზღვრის ძალზე ეფექტურ მეთოდს. წრფივ განტოლებათა სისტემის უშუალო ამოხსნის მეთოდთან შედარებით ის მნიშვნელოვნად ამცირებს დეკოდირების სირთულეს და მრავალ პრაქტიკულ შემთხვევაში გვაძლევს RS კოდების გამოყენების საშუალებას.

ახლა გადავიდეთ შეცდომების მნიშვნელობების გამოთვლაზე. აქამდე ჩვენ განვიხილავდით შემდეგი „გასაღები“ განტოლების ამოხსნის შესაძლებლობას:

$$C(x)S(x) = -T(x) \bmod x^{2t}, \quad t = \left\lfloor \frac{d-1}{2} \right\rfloor, \quad \deg C(x) > T(x),$$

სადაც $S_0, S_1, ..., S_{2t-1}$ კოეფიციენტები შეესაბამება $r(x)$ მრავალწევრის (რომელიც დამოკიდებულია მხოლოდ შეცდომების $f(x)$ მრავალწევრზე) გარდაქმნის კოორდინატებს: $A_i = A_{i+1} = ... = A_{i+2t-1}$ ყველა $A(x)$-სთვის, $A(x) \bullet -a(x) \in C$.



ნახ. 6. მესი-ბერლეკამპის ალგორითმის ბლოკ-სქემა

ცხრილი 2.

j	k	l	Δ_i	$C^{(j+1)}(x)$	$2l > j$	$B^{(j+1)}(x)$
0	0	0	$\Delta_0 = S_0 = 5$	$C^{(1)} = 1 - 5x^0 \cdot 0 = 1$	არა	$B^{(1)} = 5^{-1} = 3$
1	1	1	$\Delta_1 = S_1 + C_1^{(1)} S_0 = 5$	$C^{(2)} = 1 - 5x \cdot 3 = 1 + 6x$	კი	$B^{(2)} = B^{(1)} = 3$
2	2	1	$\Delta_2 = S_2 + C_1^{(2)} S_1 = 3 + 6 \cdot 5 = 5$	$C^{(3)} = C^{(2)} - \Delta_2 x^2 B^{(2)} = 1 + 6x - 5x^2 \cdot 3 = 1 + 6x + 6x^2$	არა	$B^{(3)} = 3(1 + 6x) = 3 + 4x$
3	1	2	$\Delta_3 = S_3 + C_1^{(3)} S_2 + C_2^{(3)} S_1 = 3 + 6 \cdot 3 + 6 \cdot 5 = 2$	$C^{(4)} = C^{(3)} - 2xB^{(3)} = 1 + 6x + 6x^2 - 2x(3 + 4x) = 1 + 5x^2$	კი	$B^{(4)} = B^{(3)}$

მიღებული შეცდომების გარდაქმნისას სინდრომი ტოლია:

$$S(x) = S_0 + S_{1x} + \ldots + S_{2t-1} x^{2t-1},$$

სადაც $S_0 = F_i, S_1 = F_{i+1}, \ldots, S_{2t-1} = F_{i+2t-1}$. ნაპოვნი $C(x)$ -ის მიხედვით ჩვენ განვსაზღვრავთ შეცდომების ადგილმდებარეობას მიღებულ კოდურ სიტყვაში. შევნიშნოთ, რომ ორობითი კოდების შემთხვევაში ჩვენ ასევე განვსაზღვრავთ შეცდომების მნიშვნელობებს, ვინაიდან $C(\alpha^i) = 0 \Rightarrow f_i = 1.$ არაორობითი კოდების შემთხვევაში ჩვენ უნდა დამატებით გამოვთვალოთ შეცდომების მნიშვნელობები $f_i \neq 0$.

ქვემოთ განვიხილავთ ორ განსხვავებულ მეთოდს.

პირველი მეთოდით განისაზღვრება $F(x)$ მრავალწევრი. ამის შემდეგ, შეცდომების მნიშვნელობები გამოითვლება $f_i = F(\alpha^i)$ ფორმულის მიხედვით. $F(x)$-ის კოეფიციენტები რეკურსიულად განისაზღვრება $C(x)F(x) = 0 \bmod(x^n - 1)$ განტოლებიდან, რომელიც შეესაბამება

$$0 = \sum_{j=0}^{e} C_j F_{n-j+l}, \quad l = 0,1,...,n-1,$$

განტოლებათა სისტემას, სადაც $C(x) = C_0 + C_1 x + ... + C_e x^e$ და $S(x) = F_i + F_{i+1}x + ... + F_{i+2t-1}x^{2t-1}$ მრავალწევრები ცნობილია. შესაბამისად, $F(x)$-ის დანარჩენი კოეფიციენტები რეკურსიულად გამოითვლება $F_{2t+i+l} = -\frac{1}{C_0}\sum_{j=1}^{e} C_j F_{2t+i-j+l}$ გან-ტოლებების მიხედვით $(l = 0,1,...,n-1-2t)$. როდესაც საბო-ლოოდ განვსაზღვრავთ $F(x)$ მრავალწევრს, შეცდომების f_i მნიშვნელობები შეცდომების $C(\alpha^i = 0)$ კოორდინატებისა-თვის გამოითვლება $f_i = F(\alpha^i)$ ტოლობის გამოყენებით.

მეორე მეთოდი გაცილებით უფრო მარტივია და ცნობილია როგორც ფორნის ალგორითმი. ამ ალგორითმის რეალიზაციისათვის ჩვენ განვიხილავთ ცვლად სიდიდეს (l), რომელიც შეესაბამება $F(x)$-ის ციკლურ წანაცვლებას: $F^{(l)}(x) = x^l F(z) \bmod(x^n - 1)$. l სიდიდე არჩეულია ისე, რომ $F_0^{(l)} = S_0$, $F_1^{(l)} = S_1, ..., F_{2t-1}^{(l)} = S_{2t-1}$. როგორც აღნიშნული იყო

ადგილი აქვს ტოლობას: $C(x)F^{(l)}(x) = 0 \bmod(x^n - 1)$, რომელიც შეიძლება ჩაიწეროს შემდეგი სახით:

$$C(x)F^{(l)}(x) = T^{(l)}(x)(x^n - 1) = -T^{(l)}(x) + x^n T^{(l)} x,$$

სადაც $\deg T^{(l)}(x) \le e-1,\ \deg C(x) = e$.

აქედან გამომდინარეობს:

$$\begin{aligned}
T_0^{(l)} &= -C_0 F_0^{(l)} && = -C_0 S_0 \\
T_1^{(l)} &= -C_0 F_1^{(l)} - C_1 F_0^{(l)} && = -C_0 S_1 - C_1 S_0 \\
T_2^{(l)} &= -C_0 S_2 - C_1 S_1 - C_2 S_0 \\
&\vdots \\
T_{e-1}^{(l)} &= -C_0 S_{e-1} - C_1 S_{e-2} - \ldots - C_{e-1} S_0.
\end{aligned}$$

$T^{(l)}(x)$-ს ეწოდება შეცდომების გამომთვლელი მრავალწევრი. იგი ტოლია შეცდომების ლოკატორი $C(x)$ მრავალწევრისა და სინდრომის $S(x)$ მრავალწევრის ნამრავლის. შეცდომების f_i მნიშვნელობები გამოითვლება ფორმულით:

$$f_i = F(\alpha^i) = x^{-l} F^{(l)}(x)\Big|_{x=\alpha^i} = x^{-l} \frac{T^{(l)}(x)(x^n - 1)}{C(x)}\Big|_{x=\alpha^i}.$$

თუ გავითვალისწინებთ იმ გარემოებას, რომ შეცდომების კოორდინატებისათვის $C(\alpha^i) = 0$, ხოლო $x^n - 1 = 0$ გალუას ველის ყველა ელემენტისათვის, მივიღებთ:

$$\frac{T^{(l)}(x)(x^n - 1)}{C(x)}\Big|_{x=\alpha^i} = \frac{0}{0}.$$

ამგვარად, ადგილი აქვს $\frac{0}{0}$ განუსაზღვრელობას და, ამიტომ, მრავალწევრის კოეფიციენტების განსაზღვრისათვის გამოვიყენოთ ლოპიტალის წესი. გამოვთვალოთ წილადის მრიცხველისა და მნიშვნელის წარმოებულები. ვინაიდან:

$$(T^{(l)}(x)(x^n-1))^{|} = T^{(l)|}(x)(x^n-1)+T^{(l)}(x)nx^{n-1} =$$
$$= T^{(l)}(x)nx^{-1}, \quad \forall \alpha^i \in GF(p),$$

შედეგად მივიღებთ f_i კოეფიციენტების შემდეგ მნიშვნელობებს:

$$f_i = x^{-l} n x^{-1} \left. \frac{T^{(l)}(x)}{C^{|}(x)} \right|_{x=\alpha^i}.$$

l მთელი რიცხვი განისაზღვრება გამოსახულებიდან: $F_0^{(l)} = S_0, F_1^{(l)} = S_1, \ldots, F_{2t-1}^{(l)} = S_{2t-1}$, და, აქედან გამომდინარე,

$$F^{(l)}(x) = x^l F(x) \bmod (x^n-1),$$

ხოლო $T^{(l)}(x)$-ის კოეფიციენტები ტოლია:

$$T_j^{(l)} - \sum_{i=0}^{j} S_{j-i} C_i, \; j = 0,1,\ldots,e-1, \text{ სადაც } \deg C(x) = e.$$

შევნიშნოთ, რომ თუ $A_i, A_{i+1}, \ldots, A_{i+k-1}$ წარმოდგენა შეესაბამება კოდური მრავალწევრის საინფორმაციო მრავალწევრს $a(x) \circ\!-\!\bullet A(x)$, მაშინ აუცილებელია შეცდომის $F(x)$ მრავალწევრის სრულად განსაზღვრა. ასევე, ვინაიდან $A(x) = e(x) - F(x)$ და მიღებული $r(x) \circ\!-\!\bullet R(x)$, ამიტომ საჭიროა $R(x)$-ის გამოთვლაც.

აღვნიშნოთ, რომ როდესაც RS კოდი აგებულია $GF(2^m)$ ველის გაფართოებაზე, გვაქვს $n = 2^m - 1$ და $n^{-1} = 1$. ამ დროს

მარტივდება $GF(2^m)$ ველის კოეფიციენტების შემცველი მრავალწევრის წარმოებულის გამოთვლა. მრავალწევრის წევრები ექსპონენტის ლუწი მაჩვენებლებით ქრება, ხოლო ექსპონენტის კენტი i მაჩვენებელი იცვლება $(i-1)$-ით (0 განიხილება როგორც ლუწი რიცხვი).

მაგალითისათვის, ავიღოთ α პრიმიტიული ელემენტი $GF(2^m)$ ველიდან და განვიხილოთ შემდეგი მრავალწევრი: $C(x)=\alpha^{i_0}+\alpha^{i_1}x+\alpha^{i_2}x^2+\alpha^{i_3}x^3$. მისი წარმოებული ტოლია $C^{|}(x)=\alpha^{i_1}+\alpha^{i_3}x^2$.

<u>მაგალითი 11.3.</u> გამოვთვალოთ შეცდომების მნიშვნელობები 11.1 მაგალითში მოცემული მიღებული მრავალწევრისათვის. სინდრომის მრავალწევრი გამოთვლილი იყო 11.2 მაგალითში და ის ტოლია $S(x)=5+5x+3x^2+3x^3$, სადაც $S_0=F_2,\dots,S_3=F_5$. მესი-ბერლეკამპის ალგორითმის გამოყენებით მივიღეთ შეცდომების ლოკატორი მრავალწევრი - $C(x)=5x^2+1$. ამის შემდეგ ჩვენ უნდა გამოვთვალოთ შეცდომების განმსაზღვრელი $T^{(-2)}(x)$ მრავალწევრი. (11.5) ფორმულის მიხედვით გვაქვს:

$$\left.\begin{aligned} T_0^{(-2)} &= -C_0S_0 && =2\\ T_1^{(-2)} &= -C_0S_1-C_1S_0 && =2 \end{aligned}\right\}\Rightarrow T^{(-2)}(x)=2x+2\,.$$

ახლა გამოვთვალოთ შეცდომათა მნიშვნელობები f_1 და f_4:

$$C^{|}(x)=3(x),\qquad T^{(-2)}(x)=2x+2,$$

$$f_{1,4} = x^{2-1} n \frac{T^{(-2)}(x)}{C^{|}(x)} \Bigg| x = 5 = \alpha^1; x = 2 = \alpha^4 = \begin{cases} 5 \cdot 6 \cdot \frac{5}{1} = 3 \\ 2 \cdot 6 \cdot \frac{6}{6} = 5 \end{cases}.$$

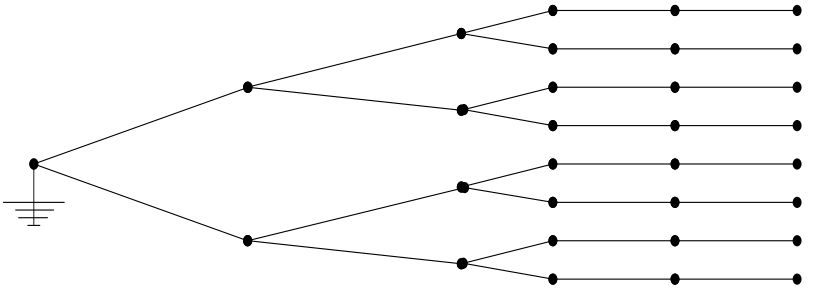
12. ხისმაგვარი კოდები

(L,T) m -ობითი ხე შეიძლება განისაზღვროს როგორც ფესვიანი ხე, რომელშიც (ა) L -ზე ნაკლებ სიღრმეზე ხის ფესვიდან თითოეული კვანძიდან გამოდის m შტო და (ბ) $(L+T)$ -ზე ნაკლებ, მაგრამ L -ის ტოლ ან მეტ სიღრმეზე ხის ფესვიდან თითოეული კვანძიდან გამოდის ერთადერთი შტო. აქ L, T და m მთელი რიცხვებია, $L \geq 1$, $T \geq 0$ და $m \geq 2$. L და T სიდიდეებს ეწოდებათ შესაბამისად ხის განშტოების სიგრძე და კუდის სიგრძე. მე-7 ნახაზზე მოცემულია ხის მაგალითი. ადვილად შეიძლება შევნიშნოთ, რომ (L,T) m -ობით ხეს აქვს m^L საბოლოო კვანძი და ეს კვანძები მოთავსებულია ხის ფესვიდან $L+T$ სიღრმეზე.

(N,R,L,T) ხისმაგვარი კოდი დისკრეტული უმეხსიერებო არხისათვის შეიძლება განისაზღვროს როგორც (L,T) 2^{NR} -ობითი ხე, რომლის თითოეულ შტოს მიკუთვნებული აქვს არხის შესასვლელი ალფაბეტის N სიმბოლო. შევნიშნოთ, რომ $m = 2^{NR}$ არის იმ შტოთა რიცხვი, რომლებიც გამოდიან თითოეული კვანძიდან ხის იმ ნაწილში, სადაც მიმდინარეობს განშტოება. R -ს ეწოდება ხისმაგვარი კოდის სიჩქარე, ხოლო სიდიდეს:

$$N_T = (T+1)N \tag{12.1}$$

ეწოდება ხისმაგვარი კოდის შემზღუდველი სიგრძე. იგი განსაზღვრავს სიმბოლოთა რაოდენობას გზის მონაკვეთზე ხის უკანასკნელი კვანძიდან, სადაც ხდება განშტოება, მის საბოლოო კვანძამდე.



ნახ. 7. $(L=3, T=2)$ ორობითი ხე

(N, R, L, T) ხისმაგვარი კოდი წარმოადგენს სპეციალური ტიპის ბლოკურ კოდს დისკრეტული უმეხსიერებო არხისათვის, რომელიც შეიცავს $m_B = m^L$ კოდურ სიტყვას სიგრძით $N_B = (L+T)N$ (სახელდობრ, N_B სიმბოლოსაგან შემდგარ m_B მიმდევრობას და თითოეული მიმდევრობა შეიძლება მიწერილ იქნას ყველა შესაძლო გზაზე ხის ფესვიდან მის საბოლოო კვანძამდე). ბლოკური კოდის სიჩქარე გამოისახება ფორმულით:

$$R_B = \frac{\log m_B}{N_B} = \frac{L}{L+T} R \tag{12.2}$$

და იგი პრაქტიკაში გამოყენებული კოდებისათვის, სადაც ჩვეულებრივ $L >> T$, დაახლოებით ისეთივეა, როგორც ხისმაგვარი კოდის სიჩქარე - R.

განვიხილოთ (N, R, L, T) ხისმაგვარი კოდების ანსამბლი დისკრეტული უმეხსიერებო არხისათვის. ამ ანსამბლში თითოეულ კოდს მიკუთვნებული აქვს ამ კოდის არჩევის ალბათობა, როდესაც არხში გადასაცემი ყველა სიმბოლო, მოთავსებული (L, T) m-ობით ხეზე, არჩეულია ერთმანეთისაგან დამოუკიდებლად, Q სიდიდის ისეთი განაწილების შესაბამისად, რომელიც ახდენს (2.2) გამოსახულების მინიმიზაციას. გამოვიყვანოთ ამ ანსამბლისათვის მაქსიმალური დამაჯერებლობით დეკოდირებისას შეცდომის საშუალო ალბათობის ზედა საზღვარი.

აღვნიშნოთ $E_i (1 \le i \le L)$ სიდიდით ხდომილება იმისა, რომ ხეზე მოთავსებული რომელიმე გზა მისული საბოლოო კვანძამდე, რომელიც განშტოვდება სწორი (გადაცემული) გზისაგან ხის ფესვიდან $L - i$ სიღრმეზე, სულ ცოტა ისევე ან მეტად შესაძლებელია „გარდაიქმნას“ მიღებულ მიმდევრობაში, როგორც სწორი გზის შესაბამისი სეგმენტი. გვაქვს:

$$P_e \le P(E_1 \cup E_2 \cup ... \cup E_L),$$

სადაც უტოლობის ნიშანი გამოყენებულია იმის გამო, რომ შეიძლება მოხდეს სწორი დეკოდირება მაშინაც კი, როდესაც მიღებული მიმდევრობის დამუშავების შემდეგ სწორ გზას აქვს ისეთივე აპოსტერიორული ალბათობა, როგორიც სხვა გზებს შორის ყველაზე უფრო მეტად სარწმუნო გზას. ადიტიური საზღვრის თანახმად:

$$P_e \le P(E_1) + P(E_2) + ... P(E_L). \qquad (12.3)$$

საერთო რაოდენობა გზებისა, რომლებიც განშტოვდებიან სწორი გზისაგან ხის ფესვიდან $L-i$ სიღრმეზე მდებარე კვანძში, ტოლია $(m-1)m^{i-1}$-ის და გზის ასეთი სეგმენტის სიგრძეა $T+i$ შტო ანუ $(T+i)N$ არხის შესასვლელი სიმბოლო. ალბათობა იმისა, რომ ერთ-ერთი ასეთი სეგმენტი უფრო მეტად შესაძლებელია „გარდაქმნილიყო“ მიღებულ მიმდევრობაში, ვიდრე სწორი გზის შესაბამისი სეგმენტი, არის შეცდომის ალბათობა ორი სიტყვისაგან შემდგარი $(T+i)N$ სიგრძის კოდისათვის. აქედან გამომდინარე, ვიყენებთ რა (2.1) ფორმულას და ადიტიურ საზღვარს, გვაქვს:

$$\overline{P(E_i)} \le (m-1)m^{i-1}2^{-(T+i)NR_0}. \qquad (12.4)$$

ამის შემდეგ თუ გავასაშუალოებთ (12.3) ფორმულაში შეცდომის ალბათობას და გამოვიყენებთ (12.4) გამოსახულებას, მივიღებთ:

$$\overline{P}_e \le (m-1)2^{-(T+1)NR_0}\sum_{i=1}^{L} m^{i-1}2^{-(i-1)NR_0}.$$

ადვილად შეიძლება შემოწმდეს, რომ თუ $L\to\infty$, მიღებული გამოსახულების მარჯვენა მხარეს მოთავსებული გეომეტრიული მწკრივის ჯამი კრებადია, როდესაც $m<2^{NR_0}$ და გვაქვს:

$$\overline{P}_e \le \frac{m-1}{1-2^{-NR_0}}2^{-(T+1)NR_0},\quad m<2^{NR_0}\quad .$$

ამის შემდეგ თუ მრიცხველში $(m-1)$ -ს შემოვსაზღვრავთ m -ით და გავიხსენებთ, რომ $m=2^{NR}$ და $N_t=(T+1)N$ საბოლოოდ მივიღებთ შემდეგ თეორემას.

თეორემა 12.1. დისკრეტულ უმეხსიერებო არხებში (N, R, L, T) ხისმაგვარი კოდების ანსამბლისათვის შეცდომის საშუალო ალბათობა მათი მაქსიმალური დამაჯერებლობით დეკოდირებისას ზემოდან შემოსაზღვრულია სიდიდით:

$$\overline{P_e} < c_t 2^{-N_t R_0}, \quad \text{როდესაც } R < R_0, \tag{12.5}$$

სადაც $c_t = \frac{1}{2^{-NR} - 2^{-NR_0}}$ ფიქსირებული R სიჩქარისთვის უმნიშვნელო მუდმივი სიდიდეა.

(12.5) ფორმულით მოცემული საზღვარი ფრიად მნიშვნელოვანია რამდენიმე გარემოების გამო: პირველი - ის გვიჩვენებს, რომ შეიძლება გამოყენებულ იქნას ხისმაგვარი კოდები ნებისმიერი სიჩქარით $0 < R < R_0$ და ყველა შემთხვევაში R_0 არის დეკოდირებისას შეცდომის ალბათობის ექსპონენტას მაჩვენებელი, ისევე როგორც მხოლოდ ორი სიტყვისთვის. მეორე - საზღვარი არ არის დამოკიდებული ხის განშტოების L სიგრძეზე და დამოკიდებულია მხოლოდ კუდის T სიგრძეზე, ვინაიდან $N_t = (T+1)N$. ამრიგად, ჩვენ შეგვიძლია ავირჩიოთ $L >> T$ და (12.2) ფორმულიდან გამომდინარე, ხისმაგვარი კოდის „ნომინალური“ სიჩქარე R ფაქტურად გავუტოლოთ ნამდვილ საინფორმაციო სიჩქარეს „ბიტებში არხის თითოეული გამოყენებისას“. მესამე - როგორც ვხედავთ ამ საზღვრის გამოყვანა არ წარმოადგენს რაიმე სირთულეს.

აღსანიშნავია, რომ ანალოგიური მსჯელობით შეიძლება ნაპოვნი იქნას $\overline{P_e}$ ალბათობის საზღვარი უფრო მაღალი სიჩქარის მქონე ხისმაგვარი კოდებისათვისაც,

$R_0 \le R < C$, სადაც C არხის გამტარუნარიანობაა [10]-[11]. მაგრამ უნდა აღინიშნოს ისიც, რომ დღეისათვის პრაქტიკაში გამოყენებული ხისმაგვარი კოდების სიჩქარე იშვიათად აღემატება R_0 -ს.

ამგვარად, (N, R, L, T) ხისმაგვარი კოდი დისკრეტული უმეხსიერებო არხისათვის შეიძლება განხილულ იქნას როგორც $(N_B = (L+T)N,\ R_B = RL/(L+T))$ ბლოკური კოდის სპეციალური ტიპი. მეორე მხრივ, (N, R) ბლოკური კოდი დისკრეტული უმეხსიერებო არხისათვის, ასევე შეიძლება განვიხილოთ როგორც $(N, R, L = 1, T = 0)$ ხისმაგვარი კოდის სპეციალური ტიპი. რომელი განსაზღვრებაა უფრო ზოგადი - ბლოკური კოდებისა თუ ხისმაგვარი კოდებისა, ეს გემოვნების საკითხია. ჩვენ ვამჯობინებთ ჩავთვალოთ ბლოკური კოდები, როგორც ხისმაგვარი კოდების კონკრეტული შემთხვევა.

13. გისოსისებრი კოდები

შემოვიტანთ რა ხისმაგვარი კოდებისათვის „მეხსიერების“ ცნებას, ჩვენ მივალთ მეტად საინტერესო კოდების კლასამდე, რომელთაც უწოდებენ „გისოსისებრ“ კოდებს. ეს სახელწოდება გამომდინარეობს იქიდან, რომ ისინი შეიძლება წარმოდგენილ იქნან გისოსისებრი დიაგრამის საშუალებით.

თავდაპირველად განვიხილოთ კოდირება (N, R, L, T) ხისმაგვარი კოდით. ვინაიდან თითოეული კვანძიდან, რომელიც მოთავსებულია L -ზე ნაკლებ სიღრმეზე ხის ფესვიდან,

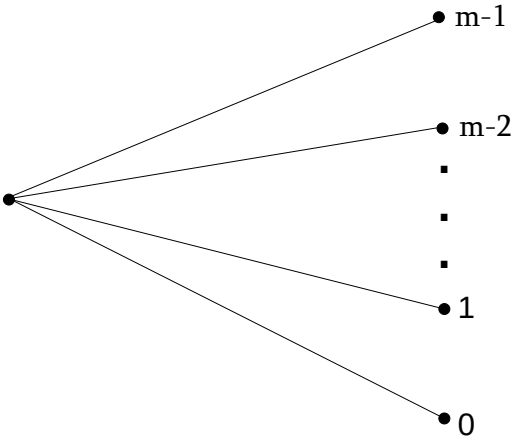
გამოდის $m = 2^{NR}$ შტო, როდესაც ვაწარმოებთ გადაადგილებას ხის სიღრმისკენ, მომდევნო შტოს განსასაზღვრად შეგვიძლია გამოვიყენოთ m-ობითი საინფორმაციო სიმბოლოების L სიგრძის მიმდევრობა $\underline{i}_0, \underline{i}_1, ..., \underline{i}_{L-1}$.

დავუშვათ, რომ საინფორმაციო სიმბოლოების ალფაბეტს აქვს სახე $\{0,1,...,m-1\}$. მე-8 ნახაზზე მოცემულია შესაბამისი არჩევის წესი. მაშინ თუ დავუბრუნდებით მე-7 ნახაზზე მოცემულ $(N, R = 1/N, L = 3, T = 2)$ ხისმაგვარ კოდს, საინფორმაციო მიმდევრობა $\underline{i}_0, \underline{i}_1, \underline{i}_2 = 0,0,0$ გამოიწვევს ხის ყველაზე ქვედა გზისათვის მიკუთვნებული კოდური, ე. ი. არხის შესასვლელი სიმბოლოების გადაცემას.

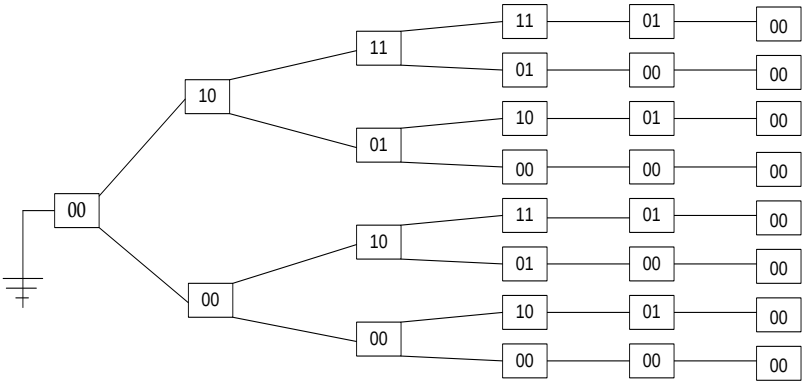
აღვნიშნოთ (L,T) m-ობითი ხის თითოეული კვანძი M წინა საინფორმაციო სიმბოლოთი, სადაც M ნებისმიერი მთელი რიცხვია დიაპაზონიდან $T \le M \le L+T$. პირობით დავუშვათ, რომ (ა) განშტოების დაწყებამდე (ე. ი. ხის ფესვამდე) გვქონდა ნულოვანი საინფორმაციო სიმბოლოები და (ბ) გზის იმ ნაწილში, რომელიც მოთავსებულია ხის ფესვიდან $L+M$ -ზე ნაკლებ, მაგრამ L ან მეტ სიღრმეზე, შესაბამისი სეგმენტის არხის შესასვლელი სიმბოლოები ყველა მიმართულებით განისაზღვრება ნულოვანი საინფორმაციო სიმბოლოებით. ამრიგად, ჩვენ მივიღეთ კვანძებაღნიშნული (L,T,M) m-ობითი ხე. მე-9 ნახაზზე მოცემულია კვანძებაღნიშნული $(L = 3,\ T = 2,\ M = 2)$ ორობითი ხე.

ახლა დავუშვათ, რომ ხისმაგვარ კოდს აქვს „მეხსიერება“ M . ეს ნიშნავს, რომ ნებისმიერი ორი კვანძიდან დაწყებული, რომლებიც ხის ფესვიდან ერთსა და იმავე

სიღრმეზეა მოთავსებული და აქვთ ერთნაირი აღნიშვნა (ე. ი. შეესაბამებათ ერთი და იგივე M წინა საინფორმაციო სიმბოლო), ხის ბოლომდე გზების დარჩენილ სეგმენტებს მიეწერებათ ერთნაირი კოდური მიმდევრობა, თუ ამ სეგმენტებს ერთნაირი საინფორმაციო მიმდევრობა შეესაბამებათ.



ნახ. 8. საინფორმაციო სიმბოლოს მიერ შტოს არჩევის წესი



ნახ. 9. $(L=3,T=2,M=2)$ კვანძებაღნიშნული ორობითი ხე

აქედან გამომდინარე, ნებისმიერ სიღრმეზე ხის ფესვიდან, ასეთი კვანძებიდან ჩვენ შეგვიძლია შევინარჩუნოთ მხოლოდ ერთი მათგანი მოცემული აღნიშვნით და მასთან გავაერთიანოთ ყველა დანარჩენი კვანძი იმავე აღნიშვნით, ე. ი. ყველა შტო, რომელიც შედიოდა დანარჩენ კვანძებში, შევიყვანოთ შენარჩუნებულ კვანძში. აღწერილი ოპერაციის შედეგად (L,T,M) კვანძებაღნიშნული m-ობითი ხის ბაზაზე მიიღება (L,T,M) m-ობითი გისოსი. მე-10 ნახაზზე მოცემულია ორი ტიპის ორობითი გისოსი. უნდა აღინიშნოს, რომ ზოგადად, (L,T,M) გისოსს აქვს m^{M-T} საბოლოო კვანძი.

ახლა შეგვიძლია განვსაზღვროთ (N,R,L,T,M) გისოსისებრი კოდი დისკრეტული უმეხსიერებო არხისათვის. ამისათვის საკმარისია (L,T,M) 2^{NR}-ობითი გისოსის თითოეულ შტოს მივაკუთვნოთ არხის N შესასვლელი სიმბოლო.

ამ მსჯელობიდან თვალნათლივ ჩანს, რომ (N,R,L,T,M) გისოსისებრი კოდი დისკრეტული უმეხსიერებო არხისათვის წარმოადგენს (N,R,L,T) ხისმაგვარი კოდის სპეციალურ ტიპს. ამიტომ სიდიდეს $N_t=(T+1)N$ კვლავ შეგვიძლია ვუწოდოთ გისოსისებრი კოდის შემზღუდველი სიგრძე, ხოლო (12.2) ფორმულა კვლავ აკავშირებს გისოსისებრი კოდის R „ნომინალურ“ სიჩქარეს და მის ნამდვილ R_B სიჩქარეს.

გამოვიყვანოთ ახლა დისკრეტული უმეხსიერებო არხებში მომუშავე გისოსისებრი კოდების ანსამბლისათვის შეც-

დომის ალბათობის ზედა საზღვარი მათი მაქსიმალური დამაჯერებლობით დეკოდირებისას. ცხადია, რომ ანსამბლის თითოეული კოდის არჩევის ალბათობა განისაზღვრება (L,T,M) m-ობით გისოსზე მოთავსებული არხის თითოეული შესასვლელი სიმბოლოს არჩევის ალბათობით. შევთანხმდეთ, რომ ეს სიმბოლოები აირჩევა ერთმანეთისაგან დამოუკიდებლად, (2.2) ფორმულაში Q-ს მამინიზებელი განაწილების შესაბამისად. ჯერ განვიხილოთ კერძო შემთხვევა - $M = T$, რომლის ბაზაზეც შემდგომში იოლად მიიღება უფრო ზოგადი შედეგი. ცხადია, რომ როდესაც $M = T$ გისოსზე გვაქვს მხოლოდ ერთი საბოლოო კვანძი, სახელდობრ, კვანძი აღნიშნული სიმბოლოთი $00...0$.

ამგვარად, გისოსზე თითოეული გზა იწყება მისი ფესვიდან და მთავრდება ამ კვანძში (ნახ. 10ა). ისევე როგორც ადრე, E_i სიდიდით აღვნიშნოთ ხდომილება იმისა, რომ გისოსზე მოთავსებული რომელიმე გზის მონაკვეთი, მისული საბოლოო კვანძამდე, სულ ცოტა ისევე ან მეტად შესაძლებელია „გარდაიქმნას“ მიღებულ მიმდევრობაში, როგორც სწორი გზის შესაბამისი სეგმენტი. ამრიგად, მაქსიმალური დამაჯერებლობით დეკოდირებისას ისევ შეიძლება გამოყენებული იქნას (12.3) ფორმულით მოცემული საზღვარი და გვაქვს:

$$P_e \le \sum_{i=1}^{L} P(E_i)\,. \qquad (13.1)$$

დავანაწილოთ E_i ხდომილება უფრო მარტივ ხდომილებად. იმისათვის, რომ სწორი გზა შეუერთდეს არასწორ გზას, მათი შესაბამისი საინფორმაციო სიმბოლოები თან-

ხვდენილი უნდა იყოს M მიმდევრობით შტოზე. აქედან გამომდინარე, სწორი გზისაგან $L-i$ სიღრმეზე (ხის ფესვიდან) მდებარე კვანძში განშტოებული გზა პირველად ისევ შეიძ-ლება შეუერთდეს სწორ გზას მხოლოდ $L+M+j-i$ სიღ-რმეზე. აღვნიშნოთ A_{ij} -ით $(1 \le j \le i)$, ხდომილება იმისა, რომ გისოსზე აღებული რომელიმე გზის სეგმენტი, რომელიც გისოსის ფესვიდან $L-i$ სიღრმეზე განშტოვდება სწორი გზისაგან და პირველად ხელახლა უერთდება მას $L+M+j-i$ სიღრმეზე მოთავსებულ კვანძში, სულ ცოტა, ისევე ან მეტად შესაძლებელია „გარდაიქმნას“ მიღებულ მიმდევრობაში, როგორც სწორი გზის შესაბამისი სეგმენტი სიგრძით $M+j$ შტო. ასეთი სეგმენტების რიცხვი არ აღემატება $m^{j-1}(m-1)$-ს, ვინაიდან თითოეული ასეთი სეგმენტის პირველი საინფორმაციო სიმბოლო განსხვავდება სწორი გზის შესაბამისი სეგმენტის პირველი საინფორმაციო სიმბოლოსაგან, ხოლო M საინფორმაციო სიმბოლო ამ ორ სეგმენტს აქვთ თანხვდენილი.

ალბათობა იმისა, რომ ერთ-ერთი ასეთი სეგმენტი უფრო მეტად შესაძლებელია „გარდაქმნილიყო“ მიღებულ მიმდევრობაში, ვიდრე სწორი გზის შესაბამისი სეგმენტი, არის ორი სიტყვისაგან შემდგარი $(M+j)N$ სიგრძის კოდის შეცდომის ალბათობა. აქედან გამომდინარე, ვიყენებთ რა (2.1) ფორმულას და ადიტიურ საზღვარს, ვიღებთ:

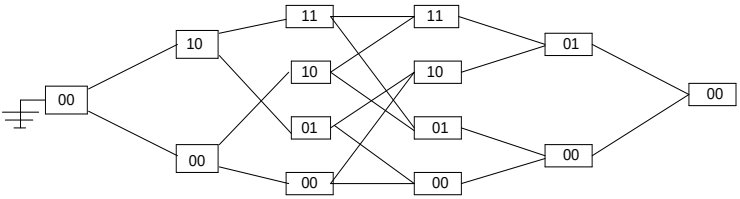
$$\overline{P(A_{ij})} \le (m-1)m^{j-1}2^{-(M+j)NR_0}. \qquad (13.2)$$

ადვილად შეიძლება შევნიშნოთ, რომ $E_i = A_{i1} \cup A_{i2} \cup ... \cup A_{ii}$ და ადიტური საზღვრის თანახმად გვაქვს:

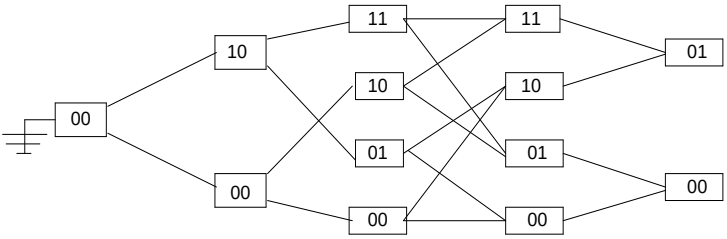
$$P(E_i) \le \sum_{j=1}^{i} P(A_{ij}) .$$

ამის შემდეგ, თუ გავასაშუალოებთ E_i ხდომილების ალბათობას და გამოვიყენებთ (13.2) ფორმულას, მივიღებთ:

$$\overline{P(E_i)} \le (m-1)2^{-(M+1)NR_0} \sum_{j=1}^{i} m^{j-1} 2^{-(j-1)R_0} .$$



ა



ბ

ნახ. 10. (ა) $(L=3, T=2, M=2)$ ორობითი გისოსი და (ბ) $(L=3, T=1, M=2)$ ორობითი გისოსი

თუ მიღებული გამოსახულების მარჯვენა მხარეს შემოვსაზღვრავთ $i \to \infty$ სიდიდით, დავინახავთ, რომ იგი ისე-

თივეა, რითაც ვისარგებლეთ (12.5) ფორმულის გამოყვანისას, ოღონდ T შეცვლილია M -ით. ამრიგად,

$$\overline{P(E_i)} \le c_t 2^{-(M+1)NR_0}, \quad R < R_0 , \tag{13.3}$$

სადაც c_t გამოითვლება (12.6) ფორმულით. საბოლოოდ, თუ გავასაშუალოებთ (13.1) გამოსახულებას და გამოვიყენებთ (13.3) ფორმულას, მივიღებთ:

$$\overline{P_e} \le c_t L 2^{-(M+1)NR_0} , \ R < R_0 , \tag{13.4}$$

რომელიც წარმოადგენს ჩვენს ძირითად საზღვარს $(N, R, L, T, M = T)$ გისოსისებრი კოდების ანსამბლისათვის მათი მაქსიმალური დამაჯერებლობით დეკოდირებისას. ხისმაგვარი კოდების შემთხვევისაგან განსხვავებით ეს საზღვარი წრფივად არის დამოკიდებული გისოსის განშტოების L სიგრძეზე.

ახლა ზემოთ მოყვანილი არგუმენტების განზოგადების საფუძველზე შევისწავლოთ (N, R, L, T, M) გისოსისებრი კოდების ანსამბლი, რომლისთვისაც მოხსნილია შეზღუდვა - $M = T$. როგორც უკვე აღინიშნა, ზოგად შემთხვევაში გისოსს აქვს m^{M-T} საბოლოო კვანძი, სადაც $T \le M \le L + T$.

დავუშვათ, რომ რომელიმე გზა, რომელიც შეიცავს გისოსის ფესვიდან $L - i$ სიღრმეზე სწორი გზისაგან განშტოებულ სეგმენტს, სულ ცოტა ისევე ან მეტად შესაძლებელია ,,გარდაქმნილიყო“ მიღებულ მიმდევრობაში, როგორც სწორი გზა. თუ ამ არასწორი გზის აღნიშნული სეგმენტი გისოსის რომელიმე კვანძში ისევ უერთდება სწორ გზას, მაშინ ასეთი შეცდომის ხდომილება შედის $T = M$ ტიპის გისოსისებრი კოდების ზემოთ მოცემულ E_i ხდომილებაში, ხოლო თუ

ასეთი სეგმენტი არ უერთდება სწორ გზას, მაშინ ამ შეცდომის ხდომილება შედის T სიგრძის კუდის მქონე ხისმაგვარი კოდების ასევე ზემოთ მოცემულ E_i ხდომილებაში.

ამგვარად, (N, R, L, T, M) გისოსისებრი კოდების ანსამბლისათვის მათი მაქსიმალური დამაჯერებლობით დეკოდირებისას $\overline{P_e}$ სიდიდე ზემოდან შემოსაზღვრულია ორი სიდიდის ჯამით, რომელთაგან პირველი მათგანი - $\overline{P_{e,1}}$ მიეკუთვნება $(N, R, L, T = M, M)$ გისოსისებრი კოდების ანსამბლს, ხოლო მეორე მათგანი - $\overline{P_{e,2}}$ მიეკუთვნება (N, R, L, T) ხისმაგვარი კოდების ანსამბლს. მაშასადამე,

$$\overline{P_e} < c_t L 2^{-(M+1)NR_0} + c_t 2^{-(T+1)NR_0},$$

რაც შეიძლება ჩაიწეროს შემდეგი სახით:

$$\overline{P_e} < c_t \left(1 + L 2^{-(M-T)NR_0}\right) 2^{-N_t R_0}, \; R < R_0. \qquad (13.5)$$

(13.5) გამოსახულება წარმოადგენს ძირითად საზღვარს (N, R, L, T, M) გისოსისებრი კოდების ანსამბლისათვის მათი მაქსიმალური დამაჯერებლობით დეკოდირებისას.

საინტერესოა აღინიშნოს, რომ (13.5) უტოლობა საშუალებას გვაძლევს, თავიდან ავიცილოთ შეცდომის ალბათობის საზღვრის არასასურველი დამოკიდებულება L სიდიდეზე, რასაც ადგილი ჰქონდა $T = M$ ტიპის გისოსისებრი კოდებისათვის (ფორმულა (13.4)).

დავუშვათ, რომ სრულდება პირობა:

$$L 2^{-(M-T)NR_0} \le 1,$$

რაც ეკვივალენტურია პირობის

$$M - T \ge \log L / (NR_0). \qquad (13.6)$$

მაშინ (13.5) ფორმულიდან მივიღებთ:

$$\overline{P_e} \le 2c_t 2^{-N_t R_0} \ . \qquad (13.7)$$

ამრიგად, თუ M სიდიდე ოდნავ მაინც აღემატება T სიდიდეს, მაშინ (13.7) ფორმულიდან გამომდინარე, $\overline{P_e}$ ალბათობა გოსოსისებრი კოდებისათვის ძალზე ახლოსაა ანალოგიურ ალბათობასთან სრული ხისმაგვარი კოდებისათვის.

ჩვენ უკვე აღვნიშნეთ, რომ (N, R, L, T, M) გისოსისებრი კოდი დისკრეტული უმეხსიერებო არხისათვის წარმოადგენს (N, R, L, T) ხისმაგვარი კოდის სპეციალურ ტიპს. ალტერნატიულად შეგვიძლია ჩავთვალოთ (N, R, L, T) ხისმაგვარი კოდი, როგორც $(N, R, L, T, M = L + T)$ გისოსისებრი კოდის სპეციალური შემთხვევა, ვინაიდან, როდესაც $M = L + T$ ასეთ შემთხვევაში გისოსი გარდაიქმნება ხედ, რომელსაც აქვს $m^{M-T} = m^L$ საბოლოო კვანძი.

ტერმინი „გისოსი“ პირველად გამოყენებული იყო დ. ფორნის მიერ [6] 10ა ნახაზზე მოცემული ტიპის გრაფების აღწერისას, რომლებითაც მან ისარგებლა ხვევადი კოდების შესასწავლად. როგორც შემდგომში ვნახავთ, ხვევადი კოდები წარმოადგენენ აქ განსაზღვრული გისოსისებრი კოდების სპეციალურ კლასს. ბოლოს უნდა აღინიშნოს, რომ (13.5) და (13.7) ფორმულებით მოცემული საზღვრები შეიძლება გავრცელებულ იქნან გისოსისებრი კოდების უფრო მაღალ სიჩქარეთა დიაპაზონზეც, $R_0 \le R < C$ [6].

14. ხვევადი კოდები

როგორც წინა პარაგრაფში აღვნიშნეთ, (N, R, L, T) ხისმაგვარი კოდების ანსამბლი დისკრეტული უმეხსიერებო არხისათვის ემთხვევა $(N, R, L, T, M = L + T)$ გისოსისებრი კოდების ანსამბლს.

ეს გარემოება საშუალებას გვაძლევს, ზოგადობის შეუზღუდავად განვიხილოთ კოდირების პროცესი მხოლოდ გისოსისებრი კოდებისათვის.

ისევე როგორც ადრე, დავუშვათ, რომ $\underline{i}_0, \underline{i}_1, ..., \underline{i}_{L-1}$ არის საკოდირებელ m-ობით საინფორმაციო სიმბოლოთა მიმდევრობა, რომლის თითოეული სიმბოლო აკონტროლებს, თუ რომელი შტო უნდა გვქონდეს კოდერის გამოსასვლელზე კვანძებს შორის გისოსის განშტოების სიგრძის შემცველ სეგმენტზე.

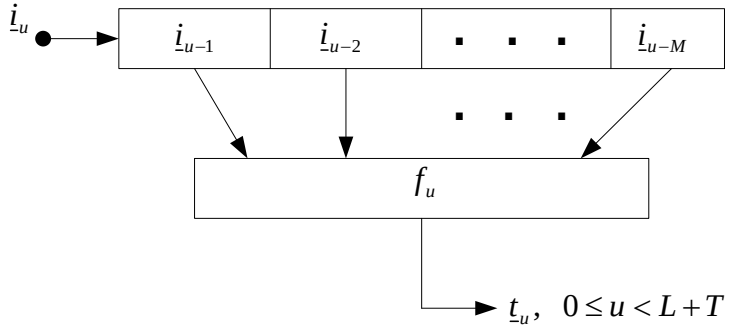
ჩვენ შეგვიძლია დავუკავშიროთ „დროის ერთეული“ თითოეულ კოდირებულ შტოს და, აქედან გამომდინარე, ჩავწეროთ კოდირებული მიმდევრობა შემდეგი სახით: $\underline{i}_0, \underline{i}_1, ..., \underline{i}_{L-1}, ..., \underline{i}_{L+T-1}$ სადაც $\underline{i}_u$ N q-ობითი სიმბოლოსაგან შემდგარი u-ურ მომენტში კოდირებული შტოა. ანალოგიურად $\underline{i}_u$ არის ,,საინფორმაციო სიმბოლო“ დროის u-ურ მომენტში, სადაც $0 \le u < L$. შევთანხმდეთ, რომ $\underline{i}_0, \underline{i}_1, ..., \underline{i}_{L-1}$ მიმდევრობას დამატებული აქვს T $\underline{0}$-ისგან შემდგარი „კუდი“, სადაც $\underline{0}$ m-ობითი ალფაბეტის რომელიმე განსაზღვრული (შეიძლება ნულოვანი) სიმბოლოა. ამრიგად, მთელი საინფორმაციო მიმდევრობა შეიძლება ჩაიწეროს შემდეგი სახით: $\underline{i}_0, \underline{i}_1, ..., \underline{i}_{L+T-1}$ სადაც $\underline{i}_u = \underline{0}$, როდესაც u მო-

თავსებულია ინტერვალში $L \le u \le L+T$. დავუშვათ აგრეთვე, რომ $\underline{i}_u = \underline{0}$ თუ $u < 0$.

ყოველივე ზემოთქმულიდან გამომდინარე, ზოგადად (N, R, L, T, M) გისოსისებრი კოდერი შეიძლება წარმოდგენილ იქნას მე-11 ნახაზზე მოცემული ბლოკ-სქემის სახით, სადაც თითოეული კვადრატი აღნიშნავს დაყოვნებას დროის ერთი ერთეულით, ხოლო მართკუთხედი აღნიშნავს ზოგად ფუნქციას, რომელიც ამყარებს დამოკიდებულებას m-ობით საინფორმაციო ალფაბეტსა და არხის შესასვლელ N სიმბოლოიან სიტყვათა მიმდევრობას შორის.

უნდა აღინიშნოს, რომ ზოგად შემთხვევაში, f_u ფუნქცია დამოკიდებულია დროზე. თუ $f_u = f$ ყველა u-სთვის დიაპაზონში $0 \le u \le L+T$, მაშინ კოდერს ვუწოდებთ მუდმივს ანუ დროზე დამოუკიდებელს.

ხვევადი (convolutional) კოდერი წარმოადგენს „წრფივ“ გისოსისებრ კოდერს. შევიმუშავოთ შესაბამისი ალგებრული სტრუქტურა, რომელიც დააკავშირებს საინფორმაციო და კოდირებულ სიმბოლოებს.



ნახ. 11. ზოგადი (N, R, L, T, M) გისოსებრი კოდერი

დავუშვათ, რომ არხის შესასვლელი სიმბოლოები აღებულია q ელემენტისაგან შემდგარი გალუას სასრული ველიდან - $GF(q)$. დავუშვათ აგრეთვე, რომ $m = q^k$ ე. ი. $\underline{i}_u$ წარმოადგენს K სიმბოლოებისაგან შემდგარ ბლოკს ელემენტებით $GF(q)$-დან. შევნიშნოთ, რომ სიდიდე

$$R = \frac{\log m}{N} = \frac{K}{N} \log q \qquad (14.1)$$

წარმოადგენს სიჩქარეს „ბიტებში არხის თითოეული გამოყენებისას“. ჩვეულებრივ, კოდირების თეორიაში განიხილავენ K / N სიდიდეს, როგორც კოდის სიჩქარეს, უფრო ზუსტად, როგორც კოდერის „უგანზომილებო სიჩქარეს“. ქვემოთ მოცემული მაგალითებისათვის შემოვიფარგლებით შემთხვევით, როდესაც $q = 2$ და, ამიტომ, $R = K / N$.

შემდგომში სიმარტივისათვის ჩვენ ჩავწერთ F ასოს $GF(q)$-ს ნაცვლად. ზოგადად (N, R, L, T, M) ხვევადი კოდერი შეიძლება წარმოდგენილ იქნას მე-11 ნახაზზე მოცემული ბლოკ-სქემის სახით, სადაც $\underline{i}_u \in F^K$, $\underline{t}_u \in F^N$, ხოლო ფუნქცია f_u, $0 \le u < L + T$, ამყარებს წრფივ დამოკიდებულებას $F^{(M+1)K}$-ის და F^N-ს შორის. f_u ფუნქცია შეიძლება ჩაიწეროს შემდეგნაირად:

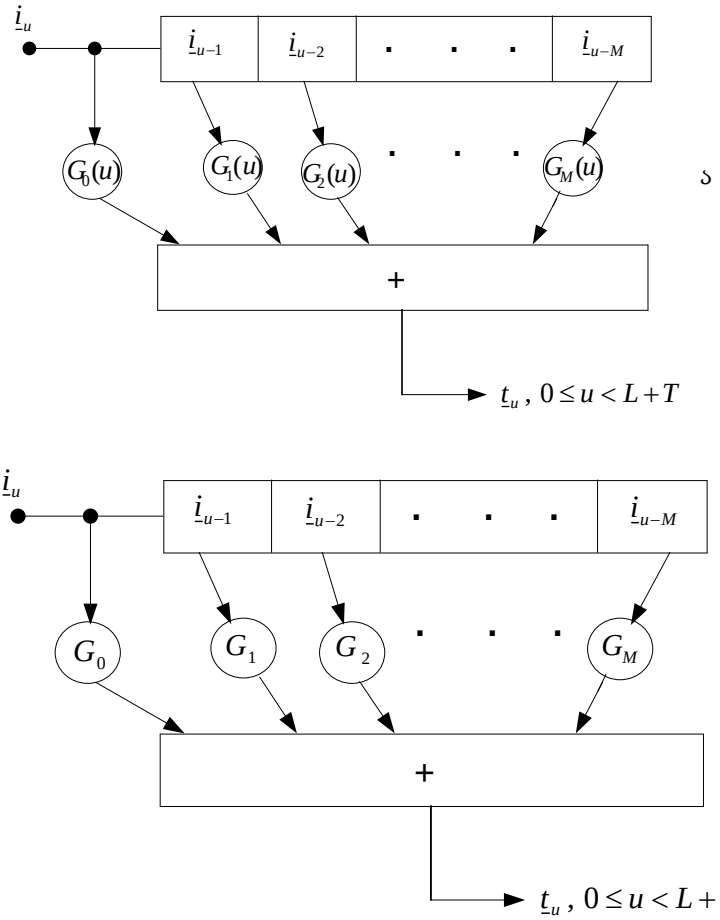
$$\underline{t}_u = \underline{i}_u G_0(u) + \underline{i}_{u-1} G_1(u) + \ldots + \underline{i}_{u-M} G_M(u), \qquad (14.2)$$

სადაც თითოეული $G_i(u)$ $K \times N$ ზომების მატრიცაა F ველზე. მუდმივი ხვევადი კოდერისათვის ეს მატრიცები არ არიან დამოკიდებული u-ზე და (14.2) ფორმულას აქვს სახე:

$$\underline{t}_u = \underline{i}_u G_0 + \underline{i}_{u-1} G_1 + \ldots + \underline{i}_{u-M} G_M \ , \qquad (14.3)$$

სადაც თითოეული $G_i(u)$ $K \times N$ მატრიცაა F ველზე.

12ა და 12ბ ნახაზებზე მოცემულია ზოგადი ხვევადი კოდერისა და ზოგადი მუდმივი ხვევადი კოდერის ბლოკ-სქემები.



ნახ. 12. (ა) ზოგადი ხვევადი კოდერი და (ბ) ზოგადი მუდმივი ხვევადი კოდერი

სახელწოდება ,,ხვევადი კოდერი“ გამომდინარეობს იმ ფაქტიდან, რომ (14.3) ფორმულით წარმოდგენილია ცნობილი მათემატიკური ოპერაცია ხვევა (convolution) საინფორმაციო მიმდევრობასა და $G_0, G_1, \ldots G_M$ მატრიცების მიმდევრობას შორის.

შემდგომი მსჯელობისათვის მოსახერხებელია ჩავწეროთ $\underline{i}_u, \underline{i}_{u+1}, \ldots, \underline{i}_v$ და $\underline{i}_u, \underline{i}_{u+1}, \ldots, \underline{i}_{v-1}$ მიმდევრობები შესაბამისად, როგორც $\underline{i}_{[u,v]}$ და $\underline{i}_{[u,v)}$, ხოლო ანალოგიური კოდური მიმდევრობები, როგორც $\underline{t}_{[u,v]}$ და $\underline{t}_{[u,v)}$. ამ აღნიშვნებიდან გამომდინარე, ზოგადი ხვევადი კოდერის მიერ (14.2) ფორმულის თანახმად შესრულებული ოპერაციები შეიძლება ჩაიწეროს შემდეგი სახით:

$$\underline{t}_{[0,L+T)} = \underline{i}_{[0,L)} \cdot \begin{pmatrix} G_0(0)\cdot G_1(1) \ldots G_M(M) & & \\ G_0(1) \ldots G_{M-1}(M) G_M(M+1) & & \\ \cdot \quad \cdot \quad \cdot & & \\ \cdot \quad \cdot \quad \cdot & & \\ \cdot \quad \cdot \quad \cdot & & \\ G_0(M) \quad G_1(M+1) \ldots & & \\ \cdot \quad \cdot & & \\ \cdot \quad \cdot & & \\ \cdot \quad \cdot & & \\ G_0(L-1) \ldots G_T(L-1) & & \end{pmatrix}, \tag{14.4}$$

სადაც თავისუფალი ადგილები მეორე მამრავლით მოცემულ სუპერმატრიცაში დაკავებულია ნულებით. (14.4) ფორმუ-

ლით მოცემულ ამ მატრიცას ეწოდება (N, R, L, T, M) ხვევადი კოდერის წარმომქმნელი მატრიცა. აღვნიშნოთ იგი $\underline{G}$ ასოთი. მუდმივი ხვევადი კოდერის შემთხვევაში მას აქვს სახე:

$$\underline{G} = \begin{pmatrix} G_0 & G_1 & \dots & G_M & & & \\ & G_0 & G_1 & \dots & G_{M-1} & G_M & \\ & & \ddots & & \ddots & & \\ & & & \ddots & & & \\ & & & & G_0 & G_1 & \dots \\ & & & & & \ddots & \\ & & & & & & \ddots \\ & & & & & & G_0 \quad G_1 \dots G_T \end{pmatrix} \tag{14.5}$$

და იგი შედგება $F = GF(q)$ ველზე განსაზღვრული $K \times N$ მატრიცებისაგან და „სტრიქონების“ რაოდენობა $\underline{G}$ სუპერ-მატრიცაში ტოლია L -ის.

15. შემთხვევითი კოდირების საზღვრები ხვევადი კოდებისათვის

ახლა შევისწავლოთ ასეთი საკითხი - ემთხვევა თუ არა ზედა საზღვრები $\overline{P_e}$ სიდიდეზე ხვევადი ხისმაგვარი და ხვევადი გისოსისებრი კოდებისათვის ანალოგიურ საზღვრებს

ხისმაგვარი და გისოსისებრი კოდების ზოგადი კლასებისათვის, რომლებიც გამოყვანილი იყო შესაბამისად მე-12 და მე-13 პარაგრაფებში. შევნიშნოთ, რომ ხვევადი (ე. ი. წრფივი) კოდების სიმეტრიულობა მოითხოვს არხის სტრუქტურის შეზღუდვას. შემდგომში განვიხილავთ მხოლოდ სიმეტრიულ არხებს და მათთვის $Q(x) = \frac{1}{q}$ (ყველა x -სთვის) წარმოადგენს მამინიზებელ განაწილებას (2.2) ფორმულაში. საბედნიეროდ, ეს შემთხვევა შეიცავს ყველა სახის ორობით შესასვლელიან არხებს, რომლებიც წარმოადგენენ განსაკუთრებულ პრაქტიკულ ინტერესს.

შემდგომში მსჯელობისათვის აუცილებელია ხვევად კოდებზე ჩავატაროთ ერთი ხელოვნური ოპერაცია, რომელიც სინამდვილეში საჭიროა ყველა ტიპის წრფივი კოდებისათვის მათი ანსამბლების ალბათური მახასიათებლების შესასწავლად. ეს ოპერაცია აუცილებელია იმ ფაქტის საჩვენებლად, რომ მოცემული ანსამბლისათვის მართებულია შემთხვევითი კოდირების გარკვეული საზღვარი, მიუხედავად იმისა, რომ ნულოვანი საინფორმაციო მიმდევრობა ყოველთვის კოდირდება ნულოვან კოდურ მიმდევრობაში (ნულოვან მიმდევრობაში იგულისხმება მხოლოდ ნულოვანი სიმბოლოებისაგან შემდგარი მიმდევრობა). ცხადია, შეუძლებელია ავირჩიოთ ხვევადი კოდების ისეთი ანსამბლი, რომელშიც კოდური სიტყვა მიიღება როგორც ,,შემთხვევითი არჩეული“ $Q(x)$ განაწილების შესაბამისად, თუ იგი შეესაბამება ნულოვან საინფორმაციო მიმდევრობას. ამ სიძნელის თავიდან ასაცილებლად კოდირების თეორიაში მიღებულია თითოეული

კოდური სიტყვისათვის, მის არხში გადაცემამდე, შემთხვევითი მიმდევრობის დამატება, ე. ი. საინფორმაციო მიმდევრობა $\underline{i}_{[0,L+T)}$ გადაიცემა, როგორც $\underline{\tau}_{[0,L+T)} = \underline{t}_{[0,L+T)} + \underline{r}_{[0,L+T)}$ მიმდევრობა. აქ $\underline{r}_{[0,L+T)}$ მიმდევრობაში შემავალი სიმბოლოები (ციფრები) აირჩევა ერთმანეთისაგან დამოუკიდებლად $Q(x) = 1/q$ (ყველა x-სთვის) განაწილების შესაბამისად. ამრიგად, ჩატარებული ოპერაცია შეიძლება ჩაიწეროს შემდეგი სახით:

$$\underline{\tau}_{[0,L+T)} = \underline{i}_{[0,L)}\underline{G} + \underline{r}_{[0,L+T)}, \qquad (15.1)$$

რაც გვიჩვენებს, რომ კონკრეტული $\underline{G}$ კოდისათვის თითოეულ კოდურ სიტყვას ემატება ერთი და იგივე შემთხვევითი მიმდევრობა. (15.1) ფორმულიდან გამომდინარეობს, რომ მოცემული $\underline{i}_{[0,L)}$ თანაბრად შესაძლებელია კოდირებული იყოს ნებისმიერ $\underline{\tau}_{[0,L+T)}$-ში, ხვევადი კოდების ანსამბლის არჩევის წესის მიუხედავად, ე. ი., იმის მიუხედავად, თუ რა ალბათური განაწილებითაა მოცემული $\underline{G}$ მატრიცები. ამრიგად, ხელოვნური ოპერაცია, რომელსაც პირობითად შეიძლება ვუწოდოთ „შემთხვევითი მიმდევრობის დამატება“, გვაძლევს კოდების ისეთი ანსამბლის მიღების საშუალებას, სადაც თითოეული კოდის ნებისმიერი კოდური მიმდევრობა (ანუ გარკვეული გზა ხეზე ან გისოსზე) შეიცავს $Q(x)$ განაწილებით მოცემულ ერთმანეთისაგან დამოუკიდებლად არჩეულ სიმბოლოებს.

ახლა გავიხსენოთ, რომ წინა პარაგრაფებში $\overline{P_e}$ სიდიდეზე საზღვრების მისაღებად ჩვენ ვსარგებლობდით მხოლოდ ორი განშტოებული გზის დამოუკიდებლობის თვისებებით.

სხვა სიტყვებით რომ ვთქვათ, ერთადერთი თვისება კოდების ანსამბლისათვის, რომელიც საშუალებას გვაძლევდა დავრწმუნებულიყავით $\overline{P_e}$ სიდიდეზე მოცემული საზღვრების სამართლიანობაში, მდგომარეობდა შემდეგში: ნებისმიერი ორი გზისათვის, რომლებიც განშტოვდება ფიქსირებულ კვანძში და შემდეგ ისევ ერთდება რომელიმე კვანძში (თუ ასეთი კვანძი საერთოდ არსებობს), კოდური სიმბოლოები გზების განშტოებულ ნაწილში არის ერთმანეთისაგან დამოუკიდებელი. კოდების ანსამბლს, რომლისთვისაც ეს პირობა სრულდება, ვუწოდოთ წყვილ-წყვილად დამოუკიდებელი. განვიხილოთ ისევ ხისმაგვარი ან გისოსისებრი კოდების ანსამბლი შემთხვევითი მიმდევრობის დამატებით. ასეთი ანსამბლი $F = GF(q)$ ველზე იქნება წყვილ-წყვილად დამოუკიდებელი მაშინ და მხოლოდ მაშინ, როდესაც ორ, ერთ რომელიმე კვანძში განშტოებული გზის შესაბამის კოდურ მიმდევრობებს შორის სხვაობა (ამ ველზე) არის მიმდევრობა, რომლის თითოეული სიმბოლო კვანძამდე, სადაც ხდება ამ ორი გზის შეერთება (თუ ასეთი კვანძი არსებობს), წარმოადგენს დამოუკიდებლად არჩეულ სიმბოლოებს $Q(x) = 1/q$ განაწილების შესაბამისად. ახლა დავუშვათ, რომ $\underline{i}_{[0,v)}$ და $\underline{i}'_{[0,v)}$ გზათა წყვილის შესაბამისი საინფორმაციო მიმდევრობებია და ეს გზები განშტოვდებიან u სიღრმეზე და არა აქვთ შეხების წერტილი სულ მცირე, v სიღრმემდე მაინც. როგორც (15.1) ფორმულიდან ჩანს, შესაბამისი კოდური მიმდევრობების სხვაობა u კვანძიდან v კვანძამდე $\underline{t}''_{[u,v)}$ იგივე კოდური მიმდევრობაა, რომელიც მიიღება საინ-

ფორმაციო მიმდევრობების სხვაობის შესაბამისი მიმდევრობის - $\underline{i}''_{[0,v)} = \underline{i}_{[0,v)} - \underline{i}'_{[0,v)}$ კოდირების შედეგად, ვინაიდან გამოკლების ოპერაცია აბათილებს შემთხვევით დამატებულ მიმდევრობას. ცხადია, რომ $\underline{i}''_{[0,u)} = \underline{0}$ და $\underline{i}''_u \neq \underline{0}$, ვინაიდან აღნიშნული ორი გზა განშტოვდება u კვანძში. ასევე ცხადია, რომ $\underline{i}''_{[u,v)} = [\underline{i}''_u, \underline{i}''_{u+1}, ..., \underline{i}''_{v-1}]$ არ შეიძლება შეიცავდეს M მიმდევრობით მოთავსებულ $\underline{0}$ -ოვან საინფორმაციო შტოს, ვინაიდან საწინააღმდეგო შემთხვევაში მოხდება ზემოთ აღნიშნული ორი გზის შეერთება v კვანძამდე (ასეთი სიტუაცია შესაძლებელია მხოლოდ ბოლო M საინფორმაციო შტოსათვის). ჩატარებული მსჯელობის საფუძველზე გვაქვს:

ლემა 15.1. (N, R, L, T, M) ხვევადი კოდების ანსამბლი შემთხვევითი მიმდევრობის დამატებით არის წყვილ-წყვილად დამოუკიდებელი მაშინ და მხოლოდ მაშინ, როდესაც u -სა და v -ს ნებისმიერი ისეთი არჩევანისათვის, რომ $0 \le u < L$ და $u < v \le L + T$ და $\underline{i}_{[0,v)}$ -ს ნებისმიერი ისეთი არჩევნისათვის, რომ $\underline{i}_j = \underline{0}$ $j < u$ -სთვის და $\underline{i}_u \neq 0$, აგრეთვე როდესაც $\underline{i}_{[u,v-1)}$ არ შეიცავს M მიმდევრობით მოთავსებულ ერთნაირ ბლოკს (ბლოკი შედგება F ველის K ელემენტებისაგან), სრულდება შემდეგი პირობა: კოდებისათვის ანსამბლში ალბათობები ისეთი წესით არის მინიჭებული, რომ რეზულტირებული $\underline{t}_{[u,v)}$ წარმოადგენს მიმდევრობას, რომლის შემადგენელი სიმბოლოები დამოუკიდებლადაა განაწილებული და თითოეულს აქვს განაწილება $Q(x) = 1/q$ ყველა x -სთვის.

მოცემული ლემის ბაზაზე შეიძლება დამტკიცდეს შემდეგი თეორემა:

თეორემა 15.1. $(N, R, L, T, M = L + T)$ მუდმივი ხვევადი კოდების ანსამბლი შემთხვევითი მიმდევრობის დამატებით არის წყვილ-წყვილად დამოუკიდებელი თუ თითოეული სიმბოლო ნებისმიერ $K \times N$ ზომების მატრიცაში - G_i, $0 \le i < L + T$, არჩეულია დამოუკიდებლად $Q(x) = 1/q$ განაწილების შესაბამისად. აქედან გამომდინარე, (12.5) ფორმულით მოცემული საზღვარი შემთხვევითი ხისმაგვარი კოდებისათვის სამართლიანია აგრეთვე ხვევადი კოდების განხილული ანსამბლისათვის, როდესაც $Q(x) = 1/q$ წარმოადგენს მამინიზებელ განაწილებას (2.2) ფორმულაში.

დამტკიცება. (14.4) და (14.5) ფორმულების საფუძველზე გვაქვს:

$$\underline{t}_{[u,L+T)} = [\underline{i}_u, \underline{i}_{u+1}, \ldots \underline{i}_{L-1}] \cdot \begin{pmatrix} G_0 & G_1 & G_2 & \ldots & G_{L+T-u-1} \\ & G_0 & G_1 & \ldots & G_{L+T-u-2} \\ & & \cdot & & \cdot \\ & & & \cdot & \cdot \\ & & & \cdot & \cdot \\ & & & G_0 & \ldots \; G_T \end{pmatrix}, \qquad (15.2)$$

სადაც $\underline{i}_j = \underline{0}$ როდესაც $j < u$. 15.1 ლემის საფუძველზე საკმარისია ვუჩვენოთ, რომ $\underline{t}_{u+j}$ სიდიდე თანაბრად ალბათურია იყოს ნებისმიერი q -ობითი N ციფრისაგან შემდგარი ბლოკი (q^N შესაძლო ბლოკიდან), როდესაც $\underline{i}_u \neq \underline{0}$, იმისდა მიუხედა-

ვად, თუ რა მნიშვნელობები აქვთ მიღებული $\underline{t}_u, \underline{t}_{u+1}, ..., \underline{t}_{u+j-1}$ სიდიდეებს (ვინაიდან $M = L + T$ შეუძლებელია გვქონდეს საინფორმაციო მონაკვეთი, რომელიც შეიცავს M მიმდევრობით მოთავსებულ $\underline{0}$-ან შტოს $\underline{i}_{[u,L+T-1)}$ მიმდევრობაში და ამიტომ 15.1 ლემაში მოყვანილი ეს ჰიპოთეზა არც კი გვჭირდება). (15.2) ფორმულიდან ჩანს, რომ $\underline{i}_u G_j$ ნამრავლი წარმოადგენს იმ ჯამის შემადგენელ კომპონენტს, რომლის საშუალებითაც მიიღება $\underline{t}_{u+j}$ და ამასთან, G_j არავითარ როლს არ ასრულებს $\underline{t}_{[u,u+j)}$-ს განსაზღვრისას. აქედან გამომდინარე, კოდების ანსამბლში, $G_0, ..., G_{j-1}$ მატრიცების და, შესაბამისად, $\underline{t}_{[u,u+j)}$-ს ნებისმიერი ფიქსირებული წესით არჩევისას, G_j თანაბარი ალბათობით შეიძლება იყოს $K \times N$ ზომების ერთ-ერთი ნებისმიერი q-ობითი მატრიცა q^{KN} შესაძლო მატრიციდან. ამრიგად, ვინაიდან $\underline{i}_u \neq \underline{0}$, შესაბამისად $\underline{i}_u G_j$ ნამრავლი თანაბრად ალბათურია იყოს ნებისმიერი q-ობითი N სიგრძის ბლოკი. ამგვარად, $\underline{t}_{u+j}$ სიდიდე, რომლის მნიშვნელობა ტოლია ჯამისა, სადაც ერთი შესაკრებია $\underline{i}_u G_j$, ხოლო მეორე შესაკრებია $G_0, ..., G_{j-1}$ მატრიცებით განსაზღვრული ფიქსირებული ვექტორი, აგრეთვე თანაბარი ალბათობით შეიძლება იყოს ნებისმიერი N სიგრძის q-ობითი ბლოკი. □

15.1. თეორემიდან გამომდინარეობს, რომ როდესაც ინფორმაციის გადაცემა წარმოებს დისკრეტულ უმეხსიერებო არხებში, ე. ი. ისეთ არხებში რომელთათვისაც $Q(x) = 1/q$

წარმოადგენს (2.2) ფორმულაში მამინიზებელ განაწილებას, მუდმივი ხვევადი კოდები, რომელთათვისაც $M = L + T$, ისეთივე „კარგები“ არიან, როგორც ზოგადი ხისმაგვარი კოდები. ბუნებრივია, იმედი უნდა ვიქონიოთ, რომ მუდმივი ხვევადი კოდები ნებისმიერი M -სთვის ისეთივე „კარგები“ იქნება, როგორც ზოგადი (N, R, L, T, M) გისოსისებრი კოდები. მაგრამ, სამწუხაროდ, აღნიშნული პრობლემა წლების განმავლობაში არსებობდა და არც ამჟამადაა სრულად გადაწყვეტილი. ახლა ჩვენ ვუჩვენებთ, რომ „დროში ცვალებადი“ ხვევადი კოდები ისეთივე „კარგებია“, როგორც ზოგადი გისოსისებრი კოდები, ხოლო შემდეგ პარაგრაფში განვიხილავთ იმ მიზეზებს, რომლებიც აყენებენ ასე ძნელად გადასაწყვეტ პრობლემას მუდმივი ხვევადი კოდების შემთხვევაში.

თეორემა 15.2. (N, R, L, T, M) ხვევადი კოდების ანსამბლი შემთხვევით მიმდევრობის დამატებით, რომელშიც თითოეული სიმბოლო ყველა $G_i(u)$ $(0 \le i \le M$ და $0 \le u < L + T)$ მატრიცაში არჩეულია დამოუკიდებლად $Q(x) = 1/q$ განაწილების შესაბამისად, არის წყვილ-წყვილად დამოუკიდებელი. აქედან გამომდინარე, (15.4) და (15.5) ფორმულებით მოცემული საზღვრები ზოგადი შემთხვევითი გისოსისებრი კოდებისათვის, რომელთაც აქვთ $(N, R, L, T, M = T)$ და (N, R, L, T, M) პარამეტრები, სამართლიანია აგრეთვე აღნიშნული ხვევადი კოდების ანსამბლისათვის, როდესაც $Q(x) = 1/q$ წარმოადგენს მამინიზებელ განაწილებას (2.2) ფორმულაში.

დამტკიცება. თავდაპირველად ჩავწეროთ ტოლობა:

$$\underline{t}_{[u,v)} = [\underline{0},...,\underline{0},\underline{i}_u,...,i_{v-1}]\cdot\begin{pmatrix} G_M(u) & & & \\ G_{M-1}(u) & G_M(u+1) & & \\ . & . & \ddots & \\ . & . & & G_M(v-1) \\ . & . & & . \\ G_1(u) & G_2(u+1) & & \\ G_0(u) & G_1(u+1) & \ddots & G_2(v-1) \\ & G_0(u+1) & \ddots & G_1(v-1) \\ & & \ddots & G_0(v-1) \end{pmatrix}, \qquad (15.3)$$

სადაც $\underline{i}_u$-ს წინ უსწრებს M $\underline{0}$-ოვანი ბლოკი, ვინაიდან $\underline{i}_j = \underline{0}$ როდესაც $j < u$. დავუშვათ, რომ $\underline{i}_u \neq \underline{0}$ და $\underline{i}_{[u,v-1)}$ არ შეიცავს M მიმდევრობით $\underline{0}$-ოვანი შტოსაგან შედგენილ შიგა მონაკვეთს. 15.1 ლემიდან გამომდინარე, თეორემის დამტკიცებისათვის საკმარისია ვუჩვენოთ, რომ $\underline{t}_{u+j}$ თანაბარი ალბათობით შეიძლება იყოს q-ობითი სიმბოლოებისაგან შემდგარი N სიგრძის ნებისმიერი ბლოკი, იმის მიუხედავად, თუ რა მნიშვნელობებს იღებენ $\underline{t}_u, \underline{t}_{u+1}, ..., \underline{t}_{u+j-1}$ ბლოკები. მაგრამ $\underline{t}_{u+j}$ ბლოკის ფორმირებაში მონაწილეობენ მხოლოდ $G_0(u+j), G_1(u+j), ..., G_M(u+j)$ მატრიცები, რომელთაგან არც ერთი მათგანი არ მოქმედებს რომელიმე წინა გადაცემულ ბლოკზე. უფრო მეტიც,

$$\underline{t}_{u+j} = \underline{i}_{u+j}G_0(u+j) + \underline{i}_{u+j-1}G_1(u+j) + ... + \underline{i}_{u+j-M}G_M(u+j)$$

და ამ გამოსახულებაში უნდა არსებობდეს სულ ცოტა ერთი საინფორმაციო შტო მაინც, რომელიც განსხვავებული იქნება

$\underline{0}$ -ოვანი შტოსაგან. დავუშვათ, რომ ასეთი შტოა $\underline{i}_{u+j-k} \neq 0$. მაშინ $\underline{t}_{[u,u+j]}$ ბლოკის ფორმირებაში მონაწილე, $G_k(u+j)$ მატრიცის გარდა, ყველა დანარჩენი მატრიცების არჩევის წესის მიუხედავად, ვექტორი $\underline{i}_{u+j-k} G_k(u+j)$ ამ მატრიცის ელემენტებისათვის ყველა შესაძლო მნიშვნელობების მინიჭებისას იღებს q -ობით N -სიგრძიან ბლოკის ყველა შესაძლო მნიშვნელობას თანაბარი ალბათობით. □

ახლა შევადაროთ ერთმანეთს 15.1 და 15.2 თეორემებით მოცემული ანსამბლების „ზომები“, სადაც სიტყვა „ზომა“ გულისხმობს იმ სიმბოლოთა რაოდენობას, რომლებიც არჩეული უნდა იქნას კოდის განსაზღვრისათვის.

15.1 თეორემით მოცემული $(N, R, L, T, M = L+T)$ მუდმივი ხვევადი კოდების ანსამბლის რომელიმე კოდის განსაზღვრისათვის ჩვენ უნდა ავირჩიოთ $(L+T)KN$ სიმბოლო G_i მატრიცების განსასაზღვრად და $(L+T)N$ სიმბოლო დამატებული შემთხვევითი მიმდევრობის განსასაზღვრად. ამიტომ ამ ანსამბლს აქვს „ზომა“ $(L+T)(K+1)N$. ვინაიდან კოდირებულ სიმბოლოთა რაოდენობა შეადგენს $(L+T)N$ -ს, შეიძლება ითქვას, რომ „თითოეულ სიმბოლოზე მოსული ზომა“ ანუ „სირთულე“ აღნიშნული ანსამბლისათვის ტოლია $K+1$-ის. ამგვარად განსაზღვრული „სირთულე“ წარმოადგენს კოდის პარამეტრთა რიცხვს, რომელიც საჭიროა კოდური სიტყვის ერთი სიმბოლოს არჩევისათვის.

15.2 თეორემით მოცემულ დროში ცვალებადი (N, R, L, T, M) ხვევადი კოდების ანსამბლის რომელიმე კოდის განსაზღვრისათვის უნდა აირჩეს $(M+1)(L+T)KN$ სიმ-

ბოლო $G_i(u)$ მატრიცების განსასაზღვრად და $(L+T)N$ სიმბოლო დამატებული შემთხვევითი მიმდევრობის განსასაზღვრად. აქედან გამომდინარე, ამ ანსამბლს აქვს „ზომა“ $[(M+1)K+1](L+T)N$ და მისი „სირთულე“ ტოლია $(M+1)K+1$-ის.

მოცემული ანსამბლის „სირთულეთა“ შედარებისას ჩნდება ასეთი კითხვა: აქვთ თუ არა „კარგ“ გისოსისებრ კოდებს მართლაც უფრო მეტი „სირთულე“, ვიდრე „კარგ“ ხისმაგვარ კოდებს, თუ ჩვენ ვერ მოვძებნეთ გისოსისებრი კოდების საზღვრების შეფასების სრულყოფილი მეთოდი? შემდეგ პარაგრაფში განხილული იქნება საზღვრების შეფასებასთან დაკავშირებული ზოგიერთი საკითხი. ახლა კი შევნიშნოთ, რომ გისოსისებრი კოდების განხილული ანსამბლი აღწევს თავის მაქსიმალურ „სირთულეს“ $(L+T)K+1$-ს, როდესაც მისი მეხსიერება აღწევს მაქსიმალურ შესაძლო მნიშვნელობას, ანუ $M=L+T-1$. მაგრამ ამ მომენტისათვის ხდება გისოსისებრი კოდის გარდაქმნა ხისმაგვარ კოდად და, ამგვარად, 15.1 თეორემის მიხედვით, მისი განსაზღვრისათვის საკმარისია $K+1$-ის ტოლი „სირთულე“.

16. მცირე „სირთულის“ მქონე „კარგი“ ხვევადი გისოსისებრი კოდების კლასი

ამ პარაგრაფში ჩვენ ვუჩვენებთ, რომ (N,R,L,T,M) ხვევადი კოდების კლასს, რომელსაც აქვს წყვილ-წყვილად დამოუკიდებლობის ათვისება, ახასიათებს გაცილებით უფრო მცირე „სირთულე“, ვიდრე 15.2 თეორემით მოცემულ

კოდთა ანსამბლს. მე-13 ნახაზზე მოცემულია ხვევადი კოდების ამ ახალი კლასის კოდერის ბლოკ-სქემა. თითოეული F_i სიდიდე წარმოადგენს მატრიცას ზომებით $K \times N$, რომლის ელემენტები მოთავსებულია $GF(q)$ ველში. კოდი განისაზღვრება F_j მატრიცებით, $0 \le j \le 2(L+T-1)$, რომლებიც თავდაპირველად მოთავსებულია კოდის დამამახსოვრებელ მოწყობილობაში და მიეწოდება ქვედა ძვრის რეგისტრს ორჯერ უფრო მაღალი სისწრაფით, ვიდრე საინფორმაციო ბლოკები (შტოები) - ზედა ძვრის რეგისტრს. თითოეული ძვრის რეგისტრის სიგრძეა M და დროის i -ურ მომენტში მის შესასვლელზე ახალი i_u საინფორმაციო შტოს და ახალი F_{2u} კოდური მატრიცის მიწოდებისას ხდება საინფორმაციო $\underline{i}_u, \underline{i}_{u-1}, ..., \underline{i}_{u-M}$ შტოების გადამრავლება $F_{2u}, F_{2u-1}, ..., F_{2u-M}$ მატრიცებზე და მიღებული ნამრავლების აჯამვა. ფორმალურად F_j მატრიცები განსაზღვრავენ დროში ცვალებად ხვევად კოდს შემდეგი წესის მიხედვით:

$$G_i(u) = F_{2u-i}, \qquad (16.1)$$

მაგრამ ეს ფორმალური ფუნქციური კავშირი ნათლად არ წარმოსახავს კოდერის რეალურ სტრუქტურას.

კოდების ამ ახალი კლასისათვის შეიძლება შემდეგი თეორემის დამტკიცება:

თეორემა 16.1. (N, R, L, T, M) ხვევადი კოდების ანსამბლი (განსაზღვრული $G_i(u) = F_{2u-i}$ მატრიცებით) შემთხვევითი მიმდევრობის დამატებით არის წყვილ-წყვილად დამოუკიდებელი, თუ თითოეული სიმბოლო ყველა F_j მატრიცაში,

$0 \le j \le 2(L+T-1)$, არჩეულია დამოუკიდებლად $Q(x)=1/q$ განაწილების შესაბამისად. აქედან გამომდინარე, შემთხვევითი გისოსისებრი კოდებისათვის (13.3) და (13.5) ფორმულებით მოცემული საზღვრები სამართლიანია აგრეთვე განხილული კოდების ანსამბლისთვისაც, როდესაც $Q(x)=1/q$ წარმოადგენს მამინიზებელ განაწილებას (2.2) ფორმულაში.

დამტკიცება. თავდაპირველად აღვნიშნოთ, რომ კოდების ამ ახალი კლასისათვის (15.3) ფორმულა იღებს სახეს:

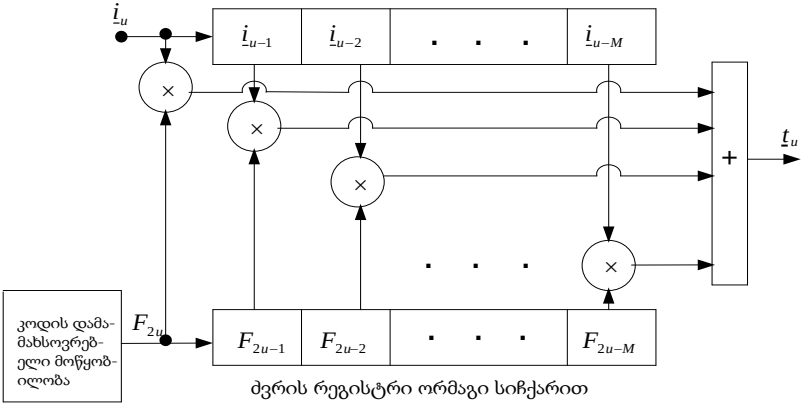
$$\underline{t}_{[u,v)} = [\underline{0},...,\underline{0},\underline{i}_u,\underline{i}_{u+1},...,\underline{i}_{v-1}]\cdot \begin{pmatrix} F_{2u-M} & & & \\ F_{2u-M+1} & F_{2u-M+2} & & \\ . & . & \ddots & \\ . & . & & F_{2v-M+2} \\ . & . & & . \\ F_{2u-1} & F_{2u} & & . \\ F_{2u} & F_{2u+1} & \ddots & F_{2v-4} \\ & F_{2u+2} & \ddots & F_{2v-3} \\ & & \ddots & F_{2v-2} \end{pmatrix}. \qquad (16.2)$$

ამასთან, როგორც ადრე, დავუშვათ, რომ $\underline{i}_u \neq 0$ და $\underline{i}_{[u,v-1)}$ არ შეიცავს შიგა მონაკვეთს, რომელიც შედგება M მიმდევრობითი $\underline{0}$-ოვანი შტოსაგან. ამრიგად, $M+1$ საინფორმაციო შტო გადამრავლებული $M+1$ არანულოვან მატრიცაზე, რომლებიც მოთავსებულია (16.2) გამოსახულების მარჯვენა სუპერმატრიცის თითოეულ „სვეტში“, არც ერთ შემთხვევაში არ მოგვცემს შედეგად მხოლოდ $\underline{0}$-ებისაგან შედგენილ მიმდევრობას. შევნიშნოთ, რომ აღნიშნულ სუპერ-მატრიცაში

მარჯვნივ დიაგონალზე გადაადგილებისას i ინდექსის მნიშვნელობა F_i მატრიცებში იზრდება (მაგალითად, $F_{2u-M}, F_{2u-M+2}, ..., F_{2v-M-2}$). კოდირებულ $\underline{t}_{u+j}$ შტოს აკონტროლებს საინფორმაციო შტოების მონაკვეთი - $\underline{i}_{[u+j-M,u+j]}$. დავუშვათ, რომ $\underline{i}_{u+k-M}$ ამ მონაკვეთზე ყველაზე უფრო მარჯვნივ მდებარე არა $\underline{0}$-ოვანი შტოა. $\underline{t}_{u+j}$ სიდიდის შემადგენელი ჯამის ფორმირებისას ეს საინფორმაციო შტო მრავლდება ისეთ F_i მატრიცაზე, რომელიც მონაწილეობას არ იღებდა არც ერთი წინა კოდირებული შტოს ფორმირებაში, ვინაიდან ეს უნდა იყოს პირველი შემთხვევა, როდესაც საინფორმაციო მიმდევრობის სუპერმატრიცის „სვეტებზე“ გადამრავლებისას, ამ სუპერ-მატრიცაში მარცხნიდან მარჯვნივ მოძრაობისას აღნიშნული F_i მატრიცა ხვდება არა $\underline{0}$-ოვან საინფორმაციო შტოს. ამრიგად, თუ გავიმეორებთ 15.2 თეორემაში მოყვანილ მსჯელობას, $\underline{t}_u, \underline{t}_{u+1}, ..., \underline{t}_{u+j-1}$ სიდიდეების მნიშვნელობების მიუხედავად, $\underline{t}_{u+j}$ თანაბარი ალბათობით შეიძლება იყოს ერთ-ერთი ნებისმიერი q-ობითი N სიგრძის ბლოკი. ამის შემდეგ, თუ გამოვიყენებთ 15.1 ლემას, საიდანაც გამომდინარეობს განხილული კოდების წყვილ-წყვილად დამოუკიდებლობის თვისება, ვრწმუნდებით, 16.1 თეორემის მართებულობაში. □

შესწავლილი ახალი ანსამბლის კოდის განსაზღვრისათვის, ჩვენ თავდაპირველად უნდა ავირჩიოთ $2(L+T)KN$ სიმბოლო F_i მატრიცებში, ხოლო შემდეგ - $(L+T)N$ სიმბოლო დამატებული შემთხვევითი მიმდევრობისათვის. ამიტომ

ამ ანსამბლს აქვს „ზომა“ $(2K+1)(L+T)N$ ანუ მისი „სირთულე“ ტოლია $2K+1$-ის. ეს მნიშვნელოვნად უკეთესია, ვიდრე დროში ცვალებადი (N,R,L,T,M) ხვევადი კოდების სრული ანსამბლის „სირთულე“ - $(M+1)K+1$, რომლებიც „კარგი“ არიან იმ გაგებით, რომ ისინი აკმაყოფილებენ მე-13 პარაგრაფში მოცემულ საზღვრებს $\overline{P_e}$ სიდიდისათვის. უნდა აღინიშნოს ისიც, რომ ახალ ანსამბლს აქვს დაახლოებით ორჯერ მეტი „სირთულე“, ვიდრე $(N,R,L,T,M=L+T)$ მუდმივი ხვევადი კოდების ანსამბლს (ამ უკანასკნელი ანსამბლის „სირთულე“ ტოლია $K+1$-ის), რომლებიც წარმოადგენენ „კარგ“ ხისმაგვარ კოდებს.



ნახ. 13. მცირე ,,სირთულის“ გისოსისებრი კოდერი

სამწუხაროდ, განხილული (N,R,L,T,M) „კარგი“ გისოსისებრი კოდების ეს კლასი არ შეიცავს ქვეკლასად მუდმივ კოდებს. თუ შევადარებთ ერთმანეთს (14.5) და (16.1)

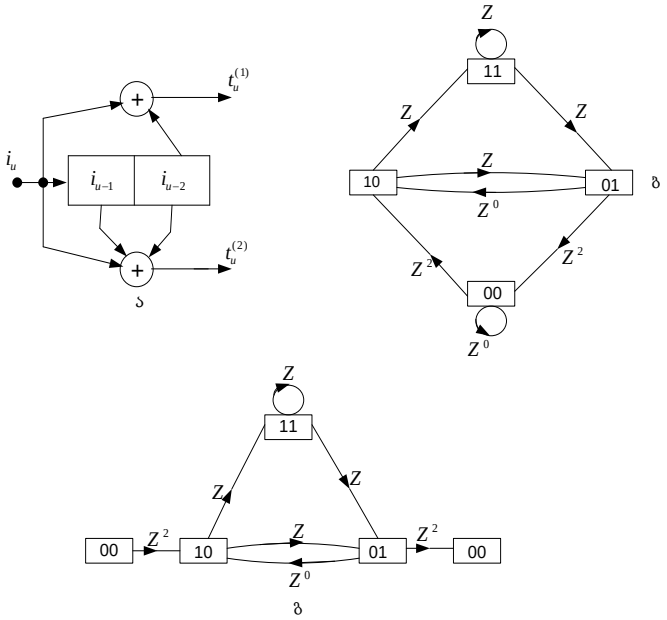
ფორმულებს, დავინახავთ, რომ ახალ კლასში შედიან მხოლოდ ის მუდმივი კოდები, რომელთათვისაც F_i მატრიცას აქვს ერთნაირი მნიშვნელობა ყველა i-სთვის და, ცხადია, რომ ამ შემთხვევაში საქმე გვაქვს ძალზე „ცუდ“ კოდებთან.

მთავარი დაბრკოლება მუდმივი ხვევადი კოდების ანსამბლის, „სიკარგის“ დამტკიცებისას მდგომარეობს იმაში, რომ ერთადერთი ცნობილი მეთოდი, რომლის საშუალებითაც შეიძლება მე-13 პარაგრაფში მოცემული $\overline{P_e}$ სიდიდის საზღვრების მიღება, ეს არის წყვილ-წყვილად დამოუკიდებლობის თვისების გამოყენება. სამწუხაროდ, მუდმივი ხვევადი კოდების ანსამბლი არ არის წყვილ-წყვილად დამოუკიდებელი, როდესაც $M < L+T$ და, ამგვარად, მუდმივი ხვევადი კოდების „სიკარგე“ არ შეიძლება დამტკიცდეს სტანდარტული მეთოდით. უნდა აღინიშნოს, რომ არ არსებობს არც იმის არგუმენტირებული დასაბუთება, რომ თუ ანსამბლი „კარგია“, აუცილებლად უნდა სრულდებოდეს წყვილ-წყვილად დამოუკიდებლობის პირობა. სავარაუდოა, რომ (N, R, L, T, M) მუდმივი ხვევადი კოდების ანსამბლი მართლაც „კარგია“. ამ მიმართულებით უკვე არსებობს ზოგიერთი შედეგი [12]. მეორე მხრივ ისიც უნდა აღინიშნოს, რომ თუ მუდმივ კოდებს მართლაც აქვს დროში ცვალებად კოდებზე უარესი მახასიათებლები, მაშინ პრაქტიკაში დღეისათვის ფართოდ დანერგილი მუდმივი ხვევადი კოდების შემცველი ინფორმაციის გადამცემი სისტემები მომავალში შეიძლება შეიცვალოს უკეთესი, დროში ცვალებად ხვევად კოდებზე დაფუძნებული სისტემებით.

17. მდგომარეობათა დიაგრამა მუდმივი ხვევადი კოდერებისათვის

მდგომარეობათა დიაგრამის გამოყენებით ა. ვიტერბიმ [13] მოძებნა მუდმივი ხვევადი კოდერების ანალიზის მეთოდი, რომელიც შეიძლება მარტივად აიხსნას მაგალითის საფუძველზე. 14ა ნახაზზე მოცემულია ორობითი ხვევადი კოდერის ბლოკ-სქემა პარამეტრებით $N = 2, K = 1$ $(R = 1/2)$ და მეხსიერებით $M = 2$. ჩვენ ჯერ არ განვსაზღვრავთ L და T სიდიდეებს, ვინაიდან ხვევადი კოდების ერთ-ერთი ღირსება მდგომარეობს იმაშიც, რომ ამ პარამეტრების არჩევა შეიძლება პრაქტიკული სიტუაციიდან გამომდინარე. კოდერის მდგომარეობათა დიაგრამა მოცემულია 14ბ ნახაზზე. იგი იგება შემდეგნაირად: თითოეულ მდგომარეობას $[i_{u-1}, i_{u-2}]$ შეესაბამება ფიქსირებული კვანძი, რომელიც აღნიშნულია ამ მდგომარეობის განმსაზღვრელი სიმბოლოებით. თითოეული კვანძიდან გამოდის მიმართული შტო, რომელიც $i_u = 0$ ან $i_u = 1$ მნიშვნელობისათვის შედის ერთერთ შესაძლო მომდევნო მდგომარეობაში, შესაბამისად, $[0, i_{u-1}]$-ში ან $[1, i_{u-1}]$ -ში. ეს შტო აღნიშნულია Z^{ω} სიდიდით, სადაც ω მოცემული გადასვლის შედეგად მიღებული კოდირებული $[t_u^{(1)}, t_u^{(2)}]$ შტოს წონაა (შემდგომში განვიხილავთ მხოლოდ ჰემინგის წონას [1], რომელიც ტოლია არანულოვან სიმბოლოთა საერთო რიცხვისა). მაგალითად, როდესაც საწყისი მდგომარეობაა $[i_{u-1}, i_{u-2}] = [1,1]$, შესასვლელი სიმბოლო $i_u = 0$ გვაძლევს კოდირებულ შტოს - $[t_u^{(1)}, t_u^{(2)}] = [1, 0]$ და

შემდეგი მდგომარეობა იქნება $[0,1]$. ვინაიდან კოდირებული შტოს ჰემინგის წონა ტოლია 1-ის, გადასვლა $[1,1]$ მდგომარეობიდან $[0,1]$ მდგომარეობაში აღინიშნება $Z^1 = Z$ სიდიდით.



ნახ.14. (ა) ოროზითი მუდმივი ხვევადი კოდერი, (ბ) მდგომარეობათა დიაგრამა და (გ) მდგომარეობათა მოდიფიცირებული დიაგრამა

განხილული მაგალითიდან ჩანს, თუ როგორ შეიძლება ნებისმიერი მუდმივი ხვევადი კოდერისათვის შესაბამისი მდგომარეობათა დიაგრამის აგება. ზოგადად, ვინაიდან ასეთ კოდერში არსებობს $(q^K)^M = q^{KM}$ განსხვავებული მდგომარე-

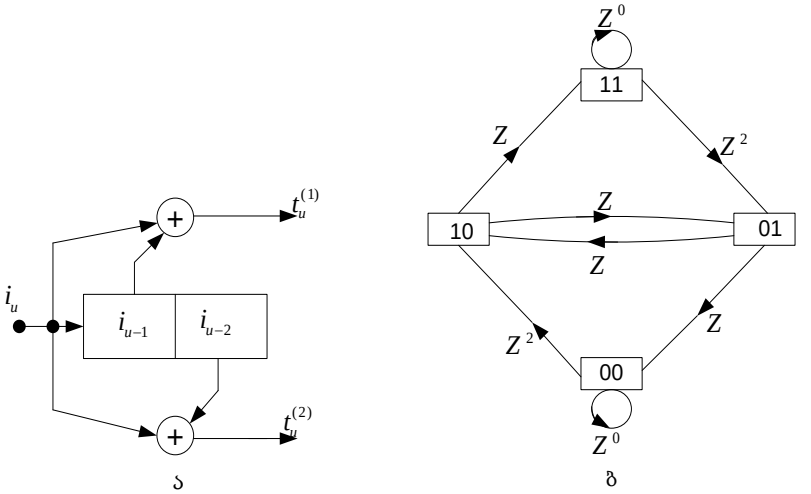
ობა, დიაგრამაზე სულ გვექნება q^{KM} კვანძი, რაც ორობით შემთხვევაშიც $(q=2)$ კი გვაძლევს მდგომარეობათა დიაგრამის მეტად რთულ სტრუქტურას. ამიტომ ასეთ დიაგრამებს იყენებენ მაშინ, როდესაც KM შედარებით მცირე სიდიდეა.

უნდა აღინიშნოს, რომ ნებისმიერი მუდმივი ხვევადი კოდერისათვის, თუ ის იმყოფება $[\underline{i}_{u-1},...,\underline{i}_{u-M}]=[\underline{0},...,\underline{0}]$ მდგომარეობაში, შესასვლელი $\underline{u}=\underline{0}$ სიმბოლო ყოველთვის გვაძლევს გამოსასვლელ $\underline{t}_u=\underline{0}$ სიმბოლოს, ამიტომ მდგომარეობათა დიაგრამა $\underline{0}$ -ოვან მდგომარეობაში ყოველთვის შეიცავს მარყუჟს, რომელიც აღნიშნულია $Z^0=1$ სიდიდით. მუდმივ ხვევად კოდერს უწოდებენ კატასტროფულს, თუ მის მდგომარეობათა დიაგრამაში მოიძებნება მიმართული შტოებისაგან შედგენილი ერთი კონტური მაინც ($\underline{0}$ -ოვან მდგომარეობაში მოთავსებული მარყუჟის გარდა), რომელიც აღნიშნული იქნება $Z^0=1$ სიდიდით, ე. ი. რომლის წარმომქმნელი ფუნქცია ტოლის 1-ის.

14ა ნახაზზე მოცემული კოდერი არაკატასტროფულია, ხოლო 15ა ნახაზზე მოცემული კოდერი - კატასტროფული, ვინაიდან $[1,1]$ მდგომარეობაში (ნახ. 15ბ) არსებობს მარყუჟი, რომლის წარმომქმნელი ფუნქცია ტოლია 1-ის. შემდგომში ჩვენ შემოვიტანთ ,,კატასტროფულობის“ ეკვივალენტურ განმარტებას, რომელიც ნათელყოფს თუ რატომ იხმარება ასეთი უცნაური ტერმინი ამგვარი კოდერებისათვის.

ამოვიღოთ $[0,0]$ მდგომარეობა და მისი შესაბამისი მარყუჟი 14ბ ნახაზზე მოცემული კოდერის მდგომარეობათა დიაგრამიდან. მივიღებთ მდგომარეობათა მოდიფიცირებულ

დიაგრამას ერთი შესასვლელი კვანძით, საიდანაც შტოები ტოვებენ $[0,0]$ მდგომარეობას, და ერთი გამოსასვლელი კვანძით, სადაც შტოები შედის $[0,0]$ მდგომარეობაში (ნახ. 14გ). ამ ოპერაციის შემდეგ ჩვენ შეგვიძლია ვიანგარიშოთ გადაცემის წარმომქმნელი ფუნქცია შესასვლელი კვანძიდან გამოსასვლელ კვანძამდე. თანაც ეს შეიძლება მოხდეს მხოლოდ იმ შემთხვევაში, როდესაც კოდერი არაკატასტროფულია და, აქედან გამომდინარე, მოდიფიცირებული დიაგრამა არ შეიცავს კონტურს ან მარყუჟს ერთეულოვანი წარმომქმნელი ფუნქციით.



ნახ.15. (ა) კატასტროფული ოროზითი მუდმივი ხვევადი კოდერი და (ბ) მისი შესაბამისი მდგომარეობათა დიაგრამა

გადაცემის წარმომქმნელი ფუნქცია $A(z)$ შეიძლება ნაპოვნი იქნას გრაფების თეორიის სტანდარტული ტექნიკის გამოყენებით, თუ გავითვალისწინებთ იმას, რომ ეს წარმომქმნელი ფუნქცია წარმოადგენს გადასასვლელ ფუნქციას ინფორმაციის ნაკადის გრაფისა ერთეულოვანი შესასვლელით. აღვნიშნოთ $[0,1]$, $[1,0]$ და $[1,1]$ მდგომარეობების შესაბამისი არასრული წარმომქმნელი ფუნქციები b, c და d ასოებით. მაშინ გვაქვს შემდეგი განტოლებათა სისტემა:

$$b = Z^2 + c, \quad d = Zb + Zd\,,\ c = Zb + Zd, \quad A(Z) = Z^2 c\,, \quad (17.1)$$

რომლის ამოხსნის შედეგად ვიპოვით:

$$A(Z) = \frac{Z^5}{1-2Z} = Z^5 + 2Z^6 + 4Z^7 + 8Z^8 + \ldots = \sum_{i=0}^{\infty} 2^i Z^{5+i}, \quad (17.2)$$

სადაც აჯამვა შეიძლება განხილულ იქნას, როგორც ხარისხოვანი მწკრივი Z ცვლადის მიმართ.

ახლა მუდმივი ხვევადი კოდერის მდგომარეობათა დიაგრამიდან მიღებულ წარმომქმნელ ფუნქციას

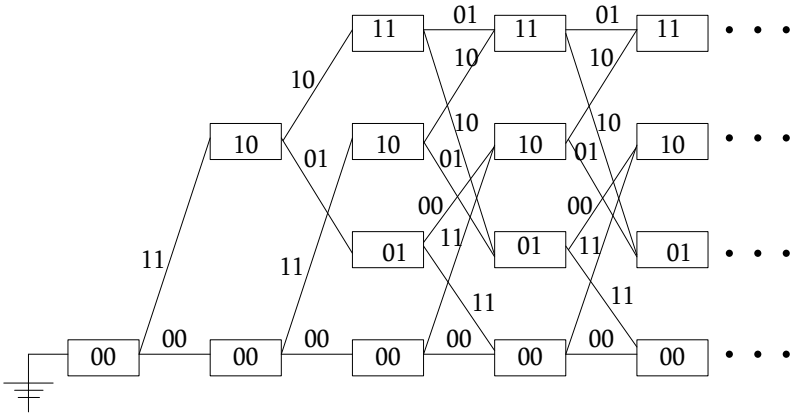
$$A(Z) = a_0 + a_1 Z + \ldots + a_i Z^i + \ldots \quad (17.3)$$

მივცეთ ინტერპრეტაცია ამ ხვევადი კოდერით განსაზღვრულ წარმომქმნელ ფუნქციას გისოსისებრი კოდის ტერმინებში, რომლისთვისაც $L \to \infty$.

მე-16 ნახაზზე წარმოდგენილია 14ა ნახაზზე მოცემული ხვევადი კოდერის შესაბამისი გისოსის საწყისი ნაწილი.

მუდმივი ხვევადი კოდერის წრფივობის გამო a_i კოეფიციენტი შეიძლება აგრეთვე განხილულ იქნას როგორც იმ გზების რიცხვი, რომლებიც განშტოვდება რომელიმე ფიქსირებული (შეიძლება არანულოვანი) გზისაგან გისოსის

ფესვურ კვანძში, აღარ სცილდება მას პირველად ხელახლა შეერთების შემდეგ და განშტოებულ სეგმენტზე მათ შორის ჰემინგის მანძილი ტოლია ზუსტად i -ს.



ნახ.16. 14ა ნახაზზე მოცემული მუდმივი კოდერით განსაზღვრული გისოსისებრი კოდის საწყისი ნაწილი, როდესაც $L \to \infty$

განვიხილოთ მოდიფიცირებულ მდგომარეობათა დიაგრამაზე (ნახ. 14გ) ნებისმიერი გზა შესასვლელი კვანძიდან გამოსასვლელ კვანძამდე. პირველი შტო შეესაბამება $[0,0]$ მდგომარეობის დატოვებას, ხოლო უკანასკნელი შტო - პირველად დაბრუნებას $[0,0]$ მდგომარეობაში. განხილული გზისათვის Z^{ϖ} სიდიდე წარმოადგენს ამ გზის ჰემინგის წონას კოდირებულ გისოსზე. (17.3) ფორმულაში a_i კოეფიციენტი ტოლია გზების რიცხვისა ჰემინგის წონით i , რომლებიც განშტოვდება ქვედა ნულოვანი გზისაგან გისოსის ფესვურ

კვანძში და აღარ სცილდებიან მას პირველად ხელახლა შეერთების შემდეგ. მაგალითად, (17.2) ფორმულიდან გამომდინარეობს, რომ გისოსზე არსებობს მხოლოდ ერთი ასეთი გზა წონით 5, ორი ასეთი გზა წონით 6, ოთხი ასეთი გზა წონით 7 და ა. შ., რაც შეიძლება მარტივად შემოწმდეს 14ა ნახაზზე მოცემული კოდერისათვის.

18. შეცდომითი დეკოდირების ალბათობის საზღვარი სპეციფიკური ხვევადი კოდერებისათვის

განვიხილოთ დისკრეტული უმეხსიერებო არხისათვის სპეციფიკური კოდი $(\underline{x}_1, \underline{x}_2)$ რომელიც შეიცავს ორ N სიგრძის სიტყვას. აღვნიშნოთ მიღებული N სიგრძის მიმდევრობა $\underline{y}$-ით, $\underline{y} = [y_1, y_2, \ldots y_N]$. მაქსიმალური დამაჯერებლობით დეკოდირებისას $\underline{x}_1$-სთვის დეკოდირების Y_1 ზონა ეს არის ყველა ისეთ $\underline{y}$-თა ერთობლიობა, რომელთათვისაც $P(\underline{y} \mid \underline{x}_1) \geq P(\underline{y} \mid \underline{x}_2)$ (გამონაკლისს შეადგენენ ისეთი $\underline{y}$-ები, რომელთათვისაც $P(\underline{y} \mid \underline{x}_1) = P(\underline{y} \mid \underline{x}_2)$ და დეკოდირების ზონად $\underline{x}_2$-სთვის შეიძლება მიჩნეულ იქნას Y_1 და Y_2 ნებისმიერი წესით). აღვნიშნოთ $P_{e|2}$ სიდიდით $\underline{x}_2$-ის გადაცემისას შეცდომით დეკოდირების ალბათობა. მართებულია შემდეგი გამოსახულება:

$$P_{e|2} \leq \sum_{\underline{y} \in Y_1} P(\underline{y} \mid \underline{x}_2). \qquad (18.1)$$

ვინაიდან $\sqrt{P(\underline{y}\,|\,\underline{x}_1)/P(\underline{y}\,|\,\underline{x}_2)} \geq 1$ ყველა $\underline{y} \in Y_1$-ისათვის, ჩვენ შეგვიძლია (18.1) ფორმულაში თითოეული შესაკრები გავამრავლოთ ამ სიდიდეზე. მივიღებთ:

$$P_{e|2} \leq \sum_{\underline{y}\in Y_1} \sqrt{P(\underline{y}\,|\,\underline{x}_1)P(\underline{y}\,|\,\underline{x}_2)}\,. \qquad (18.2)$$

გავავრცელოთ (18.2) ფორმულაში აჯამვის არე ყველა $\underline{y}$-ისათვის. მაშინ $P_{e|2}$ სიდიდე ზემოდან შემოსაზღვრულია გამოსახულებით:

$$P_{e|2} \leq \sum_{\underline{y}\in Y_1} \sqrt{P(\underline{y}\,|\,\underline{x}_1)P(\underline{y}\,|\,\underline{x}_2)}\,. \qquad (18.3)$$

ვინაიდან (18.3) ფორმულის მარჯვენა ნაწილი სიმეტრიულია $\underline{x}_1$ და $\underline{x}_2$ სიდიდეების მიმართ, ამიტომ იგივე ზედა საზღვარი სამართლიანი $P_{e|1}$ სიდიდისთვისაც, რომელიც წარმოადგენს $\underline{x}_1$-ის გადაცემისას შეცდომით დეკოდირების ალბათობას. მაგრამ ასეთ შემთხვევაში მოცემული გამოსახულება აგრეთვე უნდა წარმოადგენდეს მაქსიმალური დამაჯერებლობით დეკოდირებისას შეცდომის ალბათობის ზედა საზღვარს, იმის მიუხედავად, თუ როგორი წესითაა არჩეული კოდის ორი სიტყვა, ე. ი.

$$P_e \leq \sum_{\forall\,\underline{y}} \sqrt{P(\underline{y}\,|\,\underline{x}_1)P(\underline{y}\,|\,\underline{x}_2)}\,. \qquad (18.4)$$

ახლა, თუ გამოვიყენებთ მე-2 პარაგრაფის აღნიშვნებს, (18.4) ფორმულა გადაიწერება შემდეგი სახით:

$$P_e \leq \prod_{n=1}^{N} \sum_{y_n\in B} \sqrt{P(\underline{y}_n\,|\,\underline{x}_{1n})P(\underline{y}_n\,|\,\underline{x}_{2n})}\,. \qquad (18.5)$$

(მკითხველი ადვილად შეამჩნევს, რომ (2.1) ფორმულით მოცემული საზღვარი შეიძლება გამოყვანილ იქნას (18.5) ფორმულიდანაც, თუ ამ უკანასკნელით მოცემულ საზღვარს გავასაშუალებთ ორი კოდური სიტყვისაგან შემდგარ კოდთა ანსამბლზე).

(18.5) საზღვარი საოცრად ზუსტია დისკრეტულ უმეხსიერებო არხში ორი სიტყვისაგან შემდგარი კოდის მაქსიმალური დამაჯერებლობით დეკოდირებისას. ამის საილუსტრაციოდ დავუბრუნდეთ პირველ ნახაზზე მოცემულ ორობით სიმეტრიულ არხს. როდესაც $x_{1n} = x_{2n}$, ჯამის შესაბამისი წევრი (18.5) ფორმულაში ტოლია 1-ის, ხოლო როდესაც $x_{1n} \neq x_{2n}$ იგი ტოლია $2\sqrt{\varepsilon(1-\varepsilon)}$ -ის, სადაც ε არხის გადასასვლელი ალბათობაა. აქედან გამომდინარე, (18.5) ფორმულა იღებს სახეს:

$$P_e \leq \left[2\sqrt{\varepsilon(1-\varepsilon)}\right]^{d_H(\underline{x}_1,\underline{x}_2)} \qquad (18.6)$$

სადაც d_H აღნიშნავს ჰემინგის მანძილს ორ N სიგრძის სიტყვას შორის. $(\underline{x}_1, \underline{x}_2)$ კოდის მაქსიმალური დამაჯერებლობით დეკოდირებისას მოცემული მარტივი საზღვარი ძალზე ახლოა P_e სიდიდის ნამდვილ მნიშვნელობასთან, რაც შეიძლება იოლად შემოწმდეს კონკრეტული მაგალითების საფუძველზე.

ზოგადად, ნებისმიერი არხისათვის ორობითი შესასვლელით (არა აუცილებლად ორობითი სიმეტრიული არხისათვის) (18.5) ფორმულით მოცემულ საზღვარს აქვს შემდეგი სახე:

$$P_e \le \gamma^{d_H(\underline{x}_1,\underline{x}_2)}, \tag{18.7}$$

სადაც

$$\gamma = \sum_{y\in B}\sqrt{P(y\,|\,0)(P(y\,|\,1)}\,. \tag{18.8}$$

და $\gamma < 1$, თუ საქმე არა გვაქვს ,,უვარგის“ არხთან, ე. ი. თუ არ სრულდება პირობა $P(y\,|\,0) = (P(y\,|\,1)$ ყველა y -ისათვის.

ახლა ვუჩვენოთ, თუ როგორ შეიძლება იქნას გამოყენებული (18.7) ფორმულით მოცემული საზღვარი არაკატასტროფული მუდმივი ხვევადი კოდერების $A(z)$ წარმომქმნელ ფუნქციასთან ერთად, რომ მივიღოთ მაქსიმალური დამაჯერებლობით შეცდომით დეკოდირების ალბათობის ზედა საზღვარი ამ კოდერებით განსაზღვრული გისოსისებრი კოდისათვის. უფრო ზუსტად, ჩვენ ზემოდან შემოვსაზღვრავთ „პირველი შეცდომის მოხდენის ალბათობას“ - $P_{e,1}$, ე. ი. ალბათობას იმისა, რომ $\underline{i}_0$ ვექტორი მაქსიმალური დამაჯერებლობით დეკოდირდება არასწორად და თანაც დეკოდერი გისოსზე ირჩევს სწორი გზისაგან ფესვურ კვანძში განშტოებულ გზას. უნდა აღინიშნოს, რომ მაქსიმალური დამაჯერებლობის დეკოდერი ახდენს $\underline{i}_0$ -ის არასწორ დეკოდირებას დაწყებული ფესვური კვანძიდან მაშინ და მხოლოდ მაშინ, როდესაც რომელიმე გზა ტოვებს სწორ გზას ფესვურ კვანძში და თანაც ეს არასწორი გზა უფრო მეტად ,,ჰგავს“ მიღებულ მიმდევრობას, ვიდრე სწორი გზა. თუ აღნიშნული არასწორი გზა იმყოფება სწორი გზისაგან i -ს ტოლ ჰემინგის მანძილზე, მაშინ (18.7) ფორმულის მიხედვით, ალბათობა იმისა, რომ იგი გამოიწვევს $\underline{i}_0$ -ის არასწორ დეკოდირებას, ზემოდან შემოსა-

ზღვრულია γ^i სიდიდით. თუ სწორი გზისაგან i-ს ტოლ ჰემინგის მანძილზე მდებარე სხვა გზების რიცხვი ტოლია a_i-ის, მაშინ ადიტიური საზღვრის თანახმად ალბათობა იმისა, რომ ისინი (ერთად აღებული) გამოიწვევენ $\underline{i}_0$-ის არასწორ დეკოდირებას ზემოდან შემოსაზღრულია $a_i\gamma^i$ სიდიდით. ამრიგად, თუ მხედველობაში მივიღებთ i-ს ყველა შესაძლო მნიშვნელობას და გამოვიყენებთ კიდევ ერთ ადიტიურ საზღვარს, საბოლოოდ გვაქვს:

$$P_{e,1} \le a_0 + a_{1\cdot\gamma} + a_{2\cdot\gamma^2} + ...,$$

რაც (17.3) ფორმულის თანახმად შეიძლება ჩაიწეროს შემდეგნაირად:

$$P_{e,1} \le A(\gamma), \tag{18.9}$$

სადაც γ განსაზღვრულია (18.8) ფორმულით. ამგვარად, ჩვენ მივიღეთ ორობითი მუდმივი ხვევადი კოდერით განსაზღვრული გისოსისებრი კოდის მაქსიმალური დამაჯერებლობით დეკოდირებისას $\underline{i}_0$ ვექტორის არასწორად დეკოდირების ალბათობის ზედა საზღვარი, რომელიც დაკავშირებულია $A(z)$ ფუნქციასთან. ვინაიდან ეს საზღვარი მართებულია $L \to \infty$ შემთხვევისათვის, იგი სამართლიანი იქნება ნებისმიერი სასრულ L-ისათვისაც თუ დავუშვებთ, რომ $T = M$, ე. ი. ნებისმიერი არასწორი გზა გისოსის რომელიმე წერტილში აუცილებლად უერთდება სწორ გზას.

განვიხილოთ კონკრეტული მაგალითი. დავუშვათ, რომ ორობითი სიმეტრიული არხის გადასასვლელი ალბათობაა $\varepsilon = 0.01$. ამ არხისათვის:

$$\gamma = 2\sqrt{\varepsilon(1-\varepsilon)} \approx 0.20 .$$

(18.9) და (17.2) ფორმულების გაერთიანებით მივიღებთ, რომ 14ა ნახაზზე მოცემული მუდმივი ხვევადი კოდერით განსაზღვრული გისოსისებრი კოდისათვის $\underline{i}_0$-ის არასწორად დეკოდირების ალბათობა მაქსიმალური დამაჯერებლობის დეკოდერის გამოყენებისას ზემოდან შემოსაზღვრულია სიდიდით:

$$P_{e,1} \le (0.20)^5 /(1-2\cdot 0.20) \approx 5\cdot 10^{-4} .$$

ამრიგად, დაბრკოლებამდგრადი კოდირების საშუალებით მნიშვნელოვნადაა ამაღლებული გადაცემის სისწორე.

ლ. ვან დე მიიბერგმა შენიშნა [14], რომ (18.9) ფორმულით მოცემული საზღვარი შეიძლება გაუმჯობესდეს ოროობითი სიმეტრიული არხისათვის. ეს გამომდინარეობს იქიდან, რომ მართალია, (18.6) საზღვარი საკმაოდ ზუსტია ლუწი $d_H(\underline{x}_1, \underline{x}_2)$-სთვის, ის შეიძლება შეიცვალოს საზღვრით:

$$P_e \le \left[2\sqrt{\varepsilon(1-\varepsilon)} \right]^{d_H(\underline{x}_1, \underline{x}_2)+1},$$

თუ $d_H(\underline{x}_1, \underline{x}_2)$ კენტია. ეს გამოწვეულია იმით, რომ ორობით სიმეტრიულ არხში შეცდომით დეკოდირების ალბათობა ერთნაირია, როდესაც $d_H(\underline{x}_1, \underline{x}_2) = 2i$ და $d_H(\underline{x}_1, \underline{x}_2) = 2i-1$, ამიტომ პირველი შემთხვევა შეიძლება გამოყენებულ იქნას მეორე შემთხვევისათვისაც. აქედან გამომდინარე, ოროობითი სიმეტრიული არხისათვის ადგილი აქვს უტოლობას:

$$P_{e,1} \le a_0 + a_2\gamma^2 + a_4\gamma^4 + \ldots + a_1\gamma^2 + a_3\gamma^4 + \ldots ,$$

ანუ

$$P_{e,1} \le \frac{1}{2}\left[(1+\gamma)A(\gamma)+(1-\gamma)A(-\gamma)\right] \qquad (18.10)$$

თუ ორობით სიმეტრიულ არხში $\varepsilon = 0.01$ და კოდერის წარმომქმნელი ფუნქცია მოცემულია (17.2) ფორმულით, მაშინ (18.10) გამოსახულებიდან ვიღებთ:

$$P_{e,1} \le 2.3 \cdot 10^{-4},$$

რაც დაახლოებით ორჯერ აუმჯობესებს ადრე მიღებულ შედეგს.

დასასრულს უნდა აღინიშნოს, რომ $L \to \infty$ შემთხვევისათვის $P_{e,1}$ სიდიდე არის აგრეთვე $\underline{i}_u$ -ს არასწორად დეკოდირების ალბათობა, იმ პირობით, რომ $\underline{i}_0, \underline{i}_1, ..., \underline{i}_{u-1}$ სიტყვები ყველა სწორად იყო დეკოდირებული მაქსიმალური დამაჯერებლობის დეკოდერით. (18.9) ფორმულით მოცემული საზღვარი სამართლიანია აგრეთვე ნებისმიერი დისკრეტული უმეხსიერებო არხისათვის ორობითი შესასვლელით და იგი ხშირად გვაძლევს საკმაოდ ზუსტ საზღვარს ისეთი კოდური სისტემებისათვის, რომლებიც განხილულ არხებში იყენებენ შედარებით მოკლე კოდური შემზღუდველი სიგრძის მქონე ხვევად კოდებს და მათი მაქსიმალური დამაჯერებლობით დეკოდირების ალგორითმს.

19. მუდმივი ხვევადი კოდერების დისტანციური მახასიათებლები

ამ პარაგრაფში ჩვენ აღვწერთ მუდმივი ხვევადი კოდერების ზოგიერთ დისტანციურ მახასიათებლებს და განვსაზ-

ღვრავთ მათ მნიშვნელობას სხვადასხვა ტიპის დეკოდერების ეფექტურად ფუნქციონირებისათვის.

i -ური რიგის d_i სვეტური მანძილი განისაზღვრება როგორც ფიქსირებული ხვევადი კოდერით აგებულ $L \to \infty$ გისოსზე მინიმალური ჰემინგის მანძილი ორ კოდირებულ $\underline{t}_{[0,i]}$ გზას შორის, რომლებიც მიიღება $\underline{i}_0$ ვექტორით განსხვავებული ორი ნებისმიერ საინფორმაციო მიმდევრობისაგან - $\underline{i}_{[0,i]}$. ეკვივალენტური განმარტების მიხედვით, d_i არის ჰემინგის მანძილი $i+1$ შტოს შემცვლელ ორ გზას შორის, რომლებიც განშტოვდებიან გისოსის ფესვურ კვანძში. მუდმივი ხვევადი კოდერების წრფივობის გამო d_i ტოლია ისეთი კოდირებული $\underline{t}_{[0,i]}$ გზის ჰემინგის წონისა, რომელიც მიიღება $\underline{i}_0 \neq \underline{0}$ ვექტორის შემცველი საინფორმაციო მიმდევრობისაგან. აღვნიშნოთ მიმდევრობის ჰემინგის წონა W_H სიდიდით. თუ გამოვიყენებთ (14.4) და (14.5) ფორმულებს, შეგვიძლია ჩავწეროთ:

$$d_i = \min_{\underline{i}_0 \neq \underline{0}} W_H([\underline{i}_0, \underline{i}_1, ..., \underline{i}_i] \cdot \begin{pmatrix} G_0 & G_1 & G_2 & \dots & G_i \\ & G_0 & G_1 & \dots & G_{i-1} \\ & & & \ddots & \\ & & & & G_0 \end{pmatrix}), \quad (19.1)$$

სადაც ადრე მიღებული დაშვების თანახმად $G_j = 0$ როდესაც $j > M$ და M წარმოადგენს მუდმივი ხვევადი კოდერის მეხსიერებას.

ბუნებრივია, იბადება კითხვა, თუ რატომ ავირჩიეთ ტერმინი „სვეტური მანძილი“. ამის გასარკვევად განვიხილოთ ნახევრად უსასრულო $\underline{G}$ სუპერმატრიცა, რომელიც მიიღება (14.5) ფორმულის მარჯვენა მხარეს, როდესაც $L \to \infty$. ეს მატრიცაა:

$$\underline{G} = \begin{pmatrix} G_0 & G_1 & \dots & G_M & & & & \\ & G_0 & \dots & G_{M-1} & G_M & & & \\ & & \ddots & & & \ddots & & \\ & & & \ddots & & & \ddots & \\ & & & & \ddots & & & \ddots \\ & & & & & G_0 & G_1 \dots & G_M \\ & & & & & & \ddots & \\ & & & & & & & \ddots \end{pmatrix}. \qquad (19.2)$$

როდესაც ჩვენ ვლაპარაკობთ სუპერსტრიქონებსა და სუპერსვეტებზე, მხედველობაში გვაქვს, რომ $K \times N$ ზომის ნებისმიერი G_j მატრიცა წარმოადგენს სუპერმატრიცის ერთ ელემენტს. მაგალითად, G მატრიცის მეორე სუპერსტრიქონია $[0G_0G_1...G_{M-1}G_M\,00...]$. (19.1) ფორმულიდან გამომდინარე, d_i წარმოადგენს მინიმალური წონის კოდურ მიმდევრობას G მატრიცის იმ დარჩენილ სტრიქონთა სივრცეში, რომელიც მიიღება $\underline{G}$-ს წაკვეთის შედეგად $i+1$ სუპერსვეტის შემდეგ და რომელიც შეიცავს პირველი სუპერსტრიქონის არანულოვან მამრავლს.

(19.1) ფორმულიდან ჩანს, რომ d_i წარმოადგენს i-ს არაკლებად ფუნქციას. უფრო მეტიც, ყველა d_i ზემოდან შემოსაზღვრულია სასრული $[G_0 G_1 ... G_M]$ მატრიცის ნებისმიერი სტრიქონის ჰემინგის წონით. ასე რომ, ყოველთვის არსებობს ზღვარი:

$$d_\infty = \lim_{i \to \infty} d_i . \tag{19.3}$$

აქედან გამომდინარე,

$$d_0 \le d_1 \le d_2 \le ... \le d_i \le d_\infty . \tag{19.4}$$

გარკვეული მიზეზების გამო, რომლებიც მალე გახდება ცნობილი, d_∞ მანძილი წარმოადგენს ერთ-ერთ უმნიშვნელოვანეს პარამეტრს ხვევადი კოდებისათვის და მას უწოდებენ „თავისუფალ მანძილს“.

თუ განვიხილავთ მუდმივი ხვევადი კოდერების მდგომარეობათა დიაგრამას, d_∞ არის უსასრულოდ გრძელი არანულოვანი გზის მინიმალური წონა, რომელიც ტოვებს ნულოვან მდგომარეობას. ვინაიდან d_∞ სასრული სიდიდეა, არაკატასტროფული ხვევადი კოდერებისათვის ასეთი არანულოვანი გზა გარკვეული პერიოდის გავლის შემდეგ უბრუნდება ნულოვან მდგომარეობას და ამ მომენტიდან მოძრაობს მხოლოდ ნულოვან მარყუჟში. $A(Z)$ სიდიდის განმარტების თანახმად არაკატასტროფული კოდერებისათვის გვაქვს:

$$d_\infty = \min\{ i \mid a_i \ne 0 \}. \tag{19.5}$$

თუ დავუბრუნდებით (18.9) ფორმულით მოცემულ საზღვარს, დავინახავთ, რომ d_∞ სიდიდეს შეაქვს ყველაზე

უფრო მნიშვნელოვანი წვლილი შეცდომის P_e ალბათობის განსაზღვრისას დისკრეტულ უმეხსიერებო არხებში მუდმივი ხვევადი კოდერით წარმოდგენილი გისოსისებრი კოდის მაქსიმალური დამაჯერებლობით დეკოდირების შემთხვევაში. ჩვეულებრივ, ერთნაირი სიჩქარისა და მეხსიერების მქონე ორი მუდმივი ხვევადი კოდერიდან ის გვაძლევს ნაკლები შეცდომის P_e ალბათობას, რომლისთვისაც მეტია თავისუფალი მანძილი - d_∞. ასეთივე სიტუაციას აქვს ადგილი „თითქმის მაქსიმალური დამაჯერებლობის დეკოდირების“ სქემებისთვისაც, როგორიცაა მიმდევრობითი დეკოდირება, განხილული მომდევნო პარაგრაფებში.

პრაქტიკული რეალიზაციისათვის მიზანშეწონილი სიჩქარეებისა და არც თუ ისე მაღალი მეხსიერებისათვის დღეისათვის ნაპოვნია ბევრი საინტერესო ხვევადი კოდი, რომელთაც აქვთ თავისუფალი მანძილის მაქსიმალური შესაძლო მნიშვნელობა [15]-[17].

d_M სიდიდეს უწოდებენ მუდმივი ხვევადი კოდერის მინიმალურ მანძილს უკუკავშირით დეკოდირებისას ანუ უბრალოდ მინიმალურ მანძილს. ეს მანძილი მნიშვნელოვანია ალგებრული ტიპის დეკოდირებისათვის, რომლებიც ფუნქციონირებენ შემდეგი წესის მიხედვით:

თავდაპირველად „პირველი კოდური შემზღუდველი სიგრძის“ შესაბამისი $\underline{r}_{[0,M]}$ მიმდევრობის მიხედვით განისაზღვრება $\underline{i}_0$, შემდეგ $\underline{i}_0$-ით განპირობებული ეფექტი „აკლდება“ მიღებულ მიმდევრობას და მეორდება იგივე პროცედურა $\underline{i}_1$-ის განსასაზღვრად $\underline{r}_{[1,M+1]}$ მიმდევრობის მიხედვით და ა.

შ. არსებობს ისეთი დეკოდერი, რომელიც სწორად განსაზღვრავს $\underline{i}_0, \underline{i}_1, ..., \underline{i}_u$ მიმდევრობას, როდესაც გვაქვს t ან ნაკლები შეცდომა თითოეული კოდური შემზღუდველი სიგრძის შესაბამის $\underline{r}_{[j,M+j]}$ $(0 \le j \le u)$ მიმდევრობაში, თანაც აუცილებლად უნდა სრულდებოდეს პირობა:

$$t \le \frac{1}{2}(d_M - 1). \qquad (19.6)$$

$\underline{d} = [d_0, d_1 ... d_M]$ მიმდევრობას ეწოდება მუდმივი ხვევადი კოდერის დისტანციური პროფილი [16]. თუ $\underline{d}$ და $\underline{d}^{/}$ არის ერთნაირი სიჩქარისა და მეხსიერების მქონე ორი მუდმივი კოდერის დისტანციური პროფილები, მაშინ ამბობენ, რომ $\underline{d} > \underline{d}^{|}$, თუ $d_j > d_j^{|}$ ისეთი უმცირესი j $(0 \le j \le M)$ ინდექსისათვის, რომლისთვისაც $\underline{d}_j \ne \underline{d}_j^{|}$. კოდი უფრო უკეთესი დისტანციური პროფილით გვაძლევს ნაკლებ თანხვდენილობას იმ კოდირებულ შტოებს შორის, რომლებიც განშტოვდება ფესვურ კვანძში და ეს განსხვავება იზრდება უფრო სწრაფად (დასაწყისში მაინც) გისოსის სიღრმისაკენ მოძრაობისას. ამის გამო, კოდი, უკეთესი დისტანციური პროფილით, ზოგადად საჭიროებს უფრო ნაკლებ გამოთვლებს მიმდევრობითი დეკოდირების პროცდურისას, ვიდრე სხვა ანალოგიური კოდი.

i-ური რიგის სტრიქონული მანძილი - r_i განისაზღვრება როგორც მუდმივი ხვევადი კოდერით აგებულ $(N, R, L = i+1, T = M, M)$ გისოსზე მინიმალური ჰემინგის მანძილი ორ განსხვავებულ გზას შორის.

კოდერის წრფივობის გამო გვაქვს:

$$r_i = \min_{\underline{i}_{[0,i]} \neq \underline{0}} W_H([\underline{i}_0, \underline{i}_1, ..., \underline{i}_i] \cdot \begin{pmatrix} G_0 & G_1 & \dots & G_M & & & \\ & G_0 & G_1 & \dots & G_M & & \\ & & \ddots & & & \ddots & \\ & & & G_0 & G_1 & \dots & G_M \end{pmatrix}). \quad (19.7)$$

ამრიგად, r_i არის $\underline{G}$ სუპერმატრიცის $i+1$ სუპერსტრიქონის შემდეგ წაკვეთის შედეგად მიღებული მატრიცის სტრიქონთა არატრივიალური წრფივი კომბინაციის მინიმალური წონა. თუ დავუბრუნდებით ხვევადი კოდერის მდგომარეობათა დიაგრამას, r_i $M+i+1$ შტოსაგან შედგენილი იმ გზის მინიმალური წონაა, რომელიც ნებისმიერ კვანძში ტოვებს ნულოვან მდგომარეობას და შემდეგ ასევე ნებისმიერ კვანძში ბრუნდება ნულოვან მდგომარეობაში.

(19.7) ფორმულიდან გამომდინარეობს, რომ r_i i-ს მიმართ არაზრდადი ფუნქციაა. ვინაიდან r_i ქვემოდან შემოსაზღვრულია 0-ით, ყოველთვის არსებობს ზღვარი:

$$r_\infty = \lim_{i \to \infty} r_i \quad . \quad (19.8)$$

ამრიგად, გვაქვს

$$r_0 \geq r_1 \geq r_2 \geq ... \geq r_i \geq ... \geq r_\infty , \quad (19.9)$$

რაც ქმნის ერთგვარ კონტრასტულ სურათს (19.4) ფორმულის მიმართ. უფრო მეტიც, (19.1) და (19.7) ფორმულებიდან გამომდინარეობს, რომ

$$d_i \leq r_i$$

ყველა i-სა და j-სთვის.

(19.4) და (19.9) ფორმულიდან გამომდინარე, ადგილი აქვს შემდეგ უტოლობებს:

$$d_0 \le d_1 \le ...d_i \le ... \le d_\infty \le r_\infty \le ... \le r_i \le ... \le r_1 \le r_0 . \quad (19.10)$$

თუ გავითვალისწინებთ d_i და r_i დისტანციური მახასიათებლების განსაზღვრებებს მდგომარეობათა დიაგრამის მიმართ, ადვილად დავრწმუნდებით, რომ არაკატასტროფული მუდმივი ხვევადი კოდერებისათვის:

$$d_\infty = r_\infty . \quad (19.11)$$

ზემოთ მოყვანილი განმარტებებიდან გამომდინარე r_i განსაზღვრავს მუდმივი ხვევადი კოდერის ბაზაზე აგებული $(L = i+1, T = M)$ გისოსის შეცდომების გამასწორებელ თვისებას, რაც ნიშნავს, რომ არსებობს დეკოდერი, რომელსაც შეუძლია მთლიან კოდირებულ მიმდევრობაში გაასწოროს t ან ნაკლებშეცდომიანი ყველა კომბინაცია, თუ სრულდება პირობა:

$$t \le \frac{1}{2}(r_i - 1) . \quad (19.12)$$

უნდა აღინიშნოს, რომ r_i სტრიქონული მანძილები არ წარმოადგენენ ხვევადი კოდერებისათვის განსაკუთრებით მნიშვნელოვან პარამეტრებს, ვინაიდან: (ა) დღეისათვის არსებული საუკეთესო ხვევადი კოდერებისათვის r_∞ მნიშვნელობა მყარდება საკმაოდ სწრაფად და ხშირად $r_0 = r_\infty$; (ბ) ინფორმაციის გადამცემ სისტემებში, როგორც წესი, გამოიყენება არაკატასტროფული მუდმივი ხვევადი კოდერები, რომელთათვისაც $r_\infty = d_\infty$. ამიტომ, სტრიქონული მანძილების გამოკვლევის მთავარი მიზანი მდგომარეობს იმაში, რომ ისინი სა-

შუალებას გვაძლევენ (19.10) ფორმულის ბაზაზე გამოვთვალოთ ხვევადი კოდერების სვეტური მანძილების ზედა საზღვარი. მაგალითად, თუ განვიხილავთ 14ა ნახაზზე მოცემულ მუდმივი ხვევადი კოდერის მდგომარეობათა დიაგრამას, ადვილად დავრწმუნდებით, რომ $d_0 = 2,\ d_1 = 3,\ \ d_2 = 4,\ \ d_3 = 5$ და $r_0 = 5$. (19.10) ფორმულიდან უშუალოდ გამომდინარეობს, რომ $d_i = 5$ ნებისმიერი $i \geq 3$-სთვის და $r_i = 5$ ნებისმიერი $i \geq 0$-სთვის. ამრიგად, აღნიშნული კოდერის თავისუფალი მანძილია $d_\infty = 5$. უნდა ითქვას ისიც, რომ კატასტროფული მუდმივი ხვევადი კოდერების შემთხვევაში (19.11) ფორმულა ყოველთვის არც არის სამართლიანი. მაგალითად, 15ა ნახაზზე მოცემული კოდერის მდგომარეობათა დიაგრამიდან ვპოულობთ $d_0 = 2$ და $d_i = 3$ ნებისმიერი $i \geq 1$-სთვის, ასე რომ $d_\infty = 3$. მეორე მხრივ, $r_i = 4$ ნებისმიერი $i \geq 0$-სთვის და, ამრიგად, $r_\infty = 4 > d_\infty = 3$.

20. კატასტროფული მუდმივი ხვევადი კოდერები

ახლა ჩვენ შევეცდებით მდგომარეობათა დიაგრამაზე დაყრდნობით უფრო ღრმად ჩავწვდეთ კატასტროფული მუდმივი ხვევადი კოდერების არსს. შევნიშნოთ, რომ კატასტროფულობის მე-17 პარაგრაფში მოყვანილი განსაზღვრება ეკვივალენტურია შემდეგი განსაზღვრების: მუდმივი ხვევადი კოდერი კატასტროფულია მაშინ და მხოლოდ მაშინ, როდესაც საინფორმაციო $\underline{i}_{[0,\infty)}$ მიმდევრობა, რომლის ჰემინგის

წონაა $W_H(\underline{i}_{[0,\infty)}) = \infty$, წარმოქმნის კოდირებულ $\underline{t}_{[0,\infty)}$ მიმდევრობას წონით $W_H(\underline{t}_{[0,\infty)}) < \infty$. ამ ორი განსაზღვრების ეკვივალენტურობა ნათლად ჩანს შემდეგი დაკვირვებიდან: თუ მდგომარეობათა დიაგრამა შეიცავს კიდევ ერთ მარყუჟს ნულოვანი წონით (იმ მარყუჟის გარდა, რომელიც მოთავსებულია ნულოვან მდგომარეობაში), მაშინ საინფორმაციო $\underline{i}_{[0,\infty)}$ მიმდევრობა, რომელსაც გადაჰყავს კოდერი ამ მარყუჟთან დაკავშირებულ მდგომარეობაში და შემდეგ ამოძრავებს მხოლოდ მის გასწვრივ, წარმოქმნის სასრული წონის კოდურ მიმდევრობას, მიუხედავად იმისა, რომ თვითონ შეიძლება წარმოადგენდეს უსასრულო წონის მიმდევრობას. სამართლიანია შებრუნებული მსჯელობაც, თუ არ არსებობს ნულოვანი წონის სხვა მარყუჟი (ნულოვან მდგომარეობაში მოთავსებული მარყუჟის გარდა), მაშინ უსასრულო წონის საინფორმაციო მიმდევრობა მუდმივად ამოძრავებს კოდერს მდგომარეობათა დიაგრამაში ისე, რომ მიღებული კოდური მიმდევრობების წონა განუწყვეტლივ იზრდება და მიისწრაფვის უსასრულობისაკენ.

კატასტროფული მუდმივი ხვევადი კოდერების კიდევ ერთი დახასიათება განპირობებულია შემდეგი ფაქტით: ასეთი კოდერი წარმოადგენს K შესასვლელისა და N გამოსასვლელის მქონე წრფივ მიმდევრობის წრედს, რომლის მეხსიერება ტოლია M-ის, რაც იმას ნიშნავს, რომ წრედის გამოსასვლელი სიდიდე დამოკიდებულია მხოლოდ მოცემულ და M წინა შესასვლელ სიდიდეებზე. წრფივ მიმდევრობით წრედს სასრული შესასვლელი მეხსიერებით უწოდებენ წინაკავშირიანს და მისი გადამცემი ფუნქციის მატრიცის წევრები,

რომლებიც შეიძლება ჩაიწეროს დაყოვნების D ოპერატორის ბაზაზე, წარმოადგენენ მრავალწევრებს. მაგალითად, 14ა ნახაზზე მოცემულ $(K=1,\ N=2,\ M=2)$ მუდმივ ხვევად კოდერს აქვს $K \times N$ ზომის გადამცემი ფუნქციის მატრიცა:

$$T(D) = (1+D^2 \quad 1+D+D^2), \tag{20.1}$$

რაც იოლად შეიძლება შემოწმდეს კოდირების სქემით მოცემული წრედისათვის.

მეორე წრფივ მიმდევრობით წრედს, $T^*(D)$ გადამცემი ფუნქციით, უწოდებენ საწყისი წრფივი მიმდევრობითი წრედის ($T(D)$ გადამცემი ფუნქციით) მიმართ ინვერსიულს $(-\Delta)$ დაყოვნებით, თუ საწყისი წრედის გამოსასვლელი მიმდევრობა, გამოყენებული ინვერსიული წრედის შესასვლელ მიმდევრობად, გვაძლევს საწყისი წრედის შესასვლელ მიმდევრობას, დაყოვნებულს Δ დროის ერთეულით. ეს ნიშნავს, რომ ამ შემთხვევაში სრულდება ტოლობა:

$$T(D)T^*(D) = D^{\Delta}. \tag{20.2}$$

მაგალითად, უმარტივესი წრფივი მიმდევრობითი წრედი გადამცემი ფუნქციის მატრიცით:

$$T^*(D) = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \tag{20.3}$$

წარმოადგენს 14ა ნახაზზე წარმოდგენილი წრედის (რომლის გადამცემი ფუნქცია მოცემულია (20.1) ფორმულით) ინვერსიას $\Delta = 1$ დაყოვნებით. ანალოგიურად, წრედი გადამცემი ფუნქციით

$$T^*(D) = \begin{pmatrix} 1+D \\ D \end{pmatrix} \tag{20.4}$$

არის იმავე საწყისი წრედის ინვერსია $\Delta = 0$ დაყოვნებით, ანუ მყისი ინვერსია.

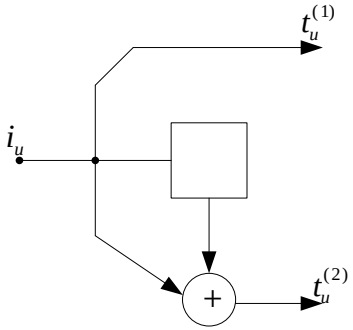
წრფივ მიმდევრობით წრედს უწოდებენ წინაკავშირიან-ინვერსიულს, თუ მას რომელიმე Δ-სთვის აქვს ინვერსია $(-\Delta)$ დაყოვნებით და ეს ინვერსია, თავის მხრივ, წარმოადგენს წინაკავშირიან წრფივ მიმდევრობით წრედს. ვინაიდან 14ა ნახაზზე მოცემული მუდმივი ხვევადი კოდერის ორივე ზემოთ მოყვანილი ინვერსია წარმოადგენს წრფივ მიმდევრობით წრედს, ეს კოდერი წინაკავშირიან-ინვერსიულია. დ. ფორნიმ უჩვენა [18], რომ წრფივი მიმდევრობითი წრედი წინაკავშირიან-ინვერსიულია მაშინ და მხოლოდ მაშინ, როდესაც არ არსებობს ისეთი უსასრულო წონის შესასვლელი მიმდევრობა, რომელიც წარმოქმნის სასრული წონის გამოსასვლელ მიმდევრობას. კატასტროფული ხვევადი კოდერის ჩვენ მიერ ზემოთ მოყვანილი მეორე დახასიათებიდან უშუალოდ გამომდინარეობს მესამე დახასიათება. სახელდობრ, მუდმივი ხვევადი კოდერი კატასტროფულია მაშინ და მხოლოდ მაშინ, როდესაც იგი წინაკავშირიან-ინვერსიულია.

ჯ. მესიმ და მ. საინმა დაამტკიცეს [19], რომ მუდმივი ხვევადი კოდერი, ან უფრო ზოგადად, წინაკავშირიანი წრფივი მიმდევრობითი წრედი წინაკავშირიან-ინვერსიულია მაშინ და მხოლოდ მაშინ, როდესაც მისი გადამცემი ფუნქციის მატრიცის $K \times K$ მინორების უდიდესი საერთო გამყოფი წარმოადგენს ერთწევრს, ე. ი. მას აქვს სახე D^i, სადაც $i \in \{0,1,\ldots\}$. აქედან გამომდინარე, 14ა ნახაზზე მოცემული კოდერი პარამეტრებით $K = 1$ და $N = 2$ არაკატასტროფულია, ვინაიდან მისი გადამცემი ფუნქციის (20.1) მატ-

რიცაში შემავალი მრავალწევრები ურთიერთმარტივია, ე. ი. მათი უდიდესი საერთო გამყოფი ტოლია 1-ის, მეორს მხრივ, 15ა ნახაზზე მოცემული მუდმივი ხვევადი კოდერის გადამცემი ფუნქციის მატრიცას აქვს სახე:

$$T(D) = (1 + D \quad 1 + D^2) \tag{20.5}$$

და თუ გავითვალისწინებთ იმას, რომ $GF(2)$ ველისთვის $1 + D^2 = (1 + D)^2$ და ამიტომ, მატრიცის წევრების უდიდესი საერთო გამყოფი ტოლია $(1 + D)$-ის, შეგვიძლია დავასკვნათ ამ კოდერის კატასტროფულობა. 15ა ნახაზზე მოცემული კატასტროფული ხვევადი კოდერის გადამცემი ფუნქციის (20.5) მატრიცის თითოეული წევრი გავყოთ $(1 + D)$-ზე. შედეგად მივიღებთ უფრო მარტივი, $M = 1$ მეხსიერების მქონე, მუდმივი ხვევადი კოდერის გადამცემი ფუნქციის მატრიცას. ამ კოდერის ბლოკ-სქემა წარმოდგენილია მე-17 ნახაზზე.



ნახ. 17. კატასტროფული მუდმივი ხვევადი კოდერის (ნახ.15ა) „ეკვივალენტური“ არაკატასტროფული მუდმივი ხვევადი კოდერი

ადვილად შეიძლება შევნიშნოთ, რომ $d_0 = 2$ და $d_i = 3$, როდესაც $i \geq 1$, ანუ ახალ კოდერს აქვს იგივე „სვეტური“ მახასიათებლები, როგორიც 15ა ნახაზზე მოცემულ კოდერს. უნდა აღინიშნოს, რომ სვეტური მანძილების შენარჩუნება არ არის შემთხვევითი და საერთოდ, ნებისმიერი კატასტროფული მუდმივი ხვევადი კოდერისათვის შეიძლება ნაპოვნი იქნას მისი „ეკვივალენტური“ არაკატასტროფული მუდმივი ხვევადი კოდერი იგივე სვეტური მანძილებით, მაგრამ ნაკლები მეხსიერებით. ეს საშუალებას გვაძლევს, თავიდან ავიცილოთ კატასტროფული კოდირება.

ჩვენ განვიხილეთ კატასტროფული მუდმივი ხვევადი კოდირების სამი ეკვივალენტური დახასიათება, მაგრამ ისინი ჯერ კიდევ არ გვაძლევენ იმის საბაბს, რომ მათ შესახებ ვიხმაროთ ტერმინი „კატასტროფული“. მეოთხე და უკანასკნელი დახასიათება (რომელიც ისტორიულად იყო „კატასტროფულობის“ თვისების პირველი დახასიათება) ნათელყოფს, თუ რატომ გახდა საჭირო ასეთი დამთრგუნველი ზედსართავი სახელის გამოყენება.

განვიხილოთ $L \to \infty$ გისოსისებრი კოდი, რომელიც გენერირდება მუდმივი ხვევადი კოდერით. დეკოდერი შეიძლება განვიხილოთ როგორც მოწყობილობა, რომელიც აფორმირებს $\underline{i}_{[0,\infty]}$ საინფორმაციო მიმდევრობის შემფასებელ $\hat{\underline{i}}_{[0,\infty)}$ მიმდევრობას. მათი სხვაობის შესაბამისი მიმდევრობა

$$\underline{\delta}_{[0,\infty)} = \hat{\underline{i}}_{[0,\infty)} - \underline{i}_{[0,\infty)} \tag{20.6}$$

წარმოადგენს დეკოდირებულ „საინფორმაციო შეცდომათა“ მიმდევრობას. ამასთან ერთად, თითოეული დეკოდერი შეიძ-

ლება განვიხილოთ აგრეთვე როგორც მოწყობილობა, რომელიც ახდენს არხში გადაცემული კოდირებებული $\underline{t}_{(0,\infty)}$ მიმდევრობის შემფასებელი $\hat{\underline{i}}_{[0,\infty)}$ მიმდევრობის გენერირებას და ეს უკანასკნელი, თავის მხრივ, შეიძლება განხილულ იქნას როგორც $\hat{\underline{i}}_{[0,\infty)}$ მიმდევრობის კოდირების შედეგი. მიმდევრობათა სხვაობა:

$$\underline{\varepsilon}_{[0,\infty)} = \hat{\underline{t}}_{[0,\infty)} - \underline{t}_{[0,\infty)} \tag{20.7}$$

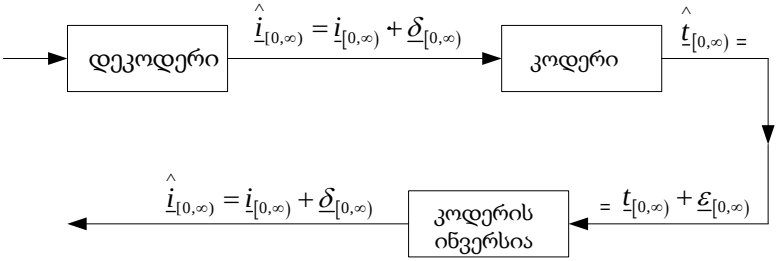
ეს არის დეკოდირებული „გადაცემის შეცდომების“ მიმდევრობა.

განვიხილოთ მე-18 ნახაზზე მოცემული დეკოდირების სქემა. აღსანიშნავია, რომ არა აქვს მნიშვნელობა, თუ როგორი სახის ინვერსიაა კოდერისათვის გამოყენებული. მუდმივი ხვევადი კოდერისა და მისი ინვერსიის წრფივობის გამო $\underline{\delta}_{[0,\infty)}$ მიმდევრობა წარმოადგენს კოდერის ინვერსიის გამოძახილს მის შესასვლელზე $\underline{\varepsilon}_{[0,\infty)}$ მიმდევრობის მიწოდების შემთხვევაში. მე-18 ნახაზზე წარმოდგენილი დეკოდერ-კოდერის ტანდემი შეიძლება განხილულ იქნას როგორც „არხის მიმდევრობის შემფასებელი“ მოწყობილობა, რომლის გამოსასვლელიც შემდგომ ინვერტირებულია $\hat{\underline{i}}_{[0,\infty)}$ მიმდევრობის მისაღებად. უნდა აღინიშნოს, რომ დღეისათვის პრაქტიკაში ფართოდ გავრცელებული დეკოდერები მუშაობენ სწორედ ამ პრინციპით: ჯერ განსაზღვრავენ არხის მიმდევრობის შემფასებელ მიმდევრობას და მხოლოდ შემდეგ ახდენენ საინფორმაციო მიმდევრობის შეფასებას.

მკითხველი დაგვეთანხმება, რომ სიტუაცია „კატასტროფულია“, როდესაც არხის მიმდევრობის შემფასებელი მიმდევრობა შეიცავს სასრული რაოდენობის შეცდომებს (ანუ $W_H(\underline{\varepsilon}_{[0,\infty)}) < \infty$), მაგრამ ეს სასრული შეცდომები იწვევენ უსასრულოდ ბევრ დეკოდირებულ „საინფორმაციო შეცდომებს“ (ანუ $W_H(\underline{\delta}_{[0,\infty)}) = \infty$). ამრიგად, ჩვენ ვამტკიცებთ, რომ მუდმივი ხვევადი კოდერი მაშინ და მხოლოდ მაშინაა კატასტროფული, თუ არხისა და დეკოდერის ყველა „რეალისტური“ წყვილისათვის, არხის არცთუ ისე სავალალო ზემოქმედებას - $W_H(\underline{\varepsilon}_{[0,\infty)}) < \infty$, თავის მხრივ, შეიძლება მოჰყვეს კატასტროფულად არასასურველი შედეგი - $W_H(\underline{\delta}_{[0,\infty)}) = \infty$. არხისა და დეკოდერის „რეალისტური“ წყვილის ქვეშ ჩვენ ვგულისხმობთ ისეთ წყვილს, რომლისთვისაც, იმის მიუხედავად, თუ რა სახის $\underline{i}_{[0,\infty]}$ იყო კოდირებული, არხმა შეიძლება მოახდინოს ისეთი ზემოქმედება, რომ დეკოდირების შედეგად მიღებული იქნება გადაწყვეტილება $\hat{\underline{i}}_{[0,\infty)} = \underline{0}$. ასეთ „რეალისტურ“ წყვილს არ მიეკუთვნება, მაგალითად, „უშეცდომო“ ორობითი სიმეტრიული არხები ნულოვანი გადასასვლელი ალბათობით, აგრეთვე ისეთი დეკოდერები, რომლებიც არასდროს არ იძლევა $\underline{0}$ -ოვან შეფასებას, მაგრამ ასეთია პრაქტიკაში არსებული „რეალური“ არხის და „გონიერი“ დეკოდერის უმეტესი სისტემები.

ჯერ ვაჩვენოთ შებრუნებული მტკიცების სამართლიანობა. განვიხილოთ არაკატასტროფული მუდმივი ხვევადი კოდერი და, ვინაიდან არა აქვს მნიშვნელობა გამოყენებული

ინვერსიის სახეს, დავუშვათ, რომ მას აქვს მე-18 ნახაზზე მოცემული ინვერსია. ჩავთვალოთ, რომ $W_H(\underline{\varepsilon}_{[0,\infty)}) < \infty$ და ისევე, როგორც ადრე, $\underline{\varepsilon}_{[0,\infty)}$-ის გამოძახილი წინაკავშირიან ინვერსიაზე აღვნიშნოთ $\underline{\delta}_{[0,\infty)}$-ით. ვინაიდან წინაკავშირიან ინვერსიას აქვს სასრული შესასვლელი მეხსიერება, დროის სასრული მონაკვეთის გავლის შემდეგ მის გამოსასვლელზე გვექნება მხოლოდ $\underline{0}$-ოვანი მიმდევრობა და ეს გამოწვეულია $\underline{\varepsilon}_{[0,\infty)}$ მიმდევრობაში, დროის გარკვეული მომენტიდან დაწყებული, მხოლოდ ნულოვანი სიმბოლოების არსებობით. ამგვარად, ვასკვნით, რომ $W_H(\underline{\delta}_{[0,\infty)}) < \infty$, რაც ადასტურებს შებრუნებული მტკიცების სამართლიანობას.



ნახ. 18. მუდმივი ხვევადი კოდერის შესაბამისი დეკოდერის კანონიკური დანაწევრება

ახლა განვიხილოთ კატასტროფული მუდმივი ხვევადი კოდერი. მაშინ არსებობს შესასვლელი $\underline{i}_{[0,\infty]}$ მიმდევრობა წონით $W_H(\underline{i}_{[0,\infty)}) = \infty$ რომლის მიხედვითაც ფორმირდება გადა-

საცემი $\underline{i}_{[0,\infty]}$ მიმდევრობა წონით $W_H(\underline{t}_{[0,\infty)}) < \infty$. დავუშვათ, რომ არხის მიერ გადაცემულ მიმდევრობაზე ზემოქმედების შედეგად ადგილი აქვს ტოლობას $\hat{\underline{i}}_{[0,\infty)} = \underline{0}$, რაც, თავის მხრივ, გულისხმობს, რომ სრულდება ტოლობა: $\underline{t}_{[0,\infty)} = 0$. ამრიგად, გვაქვს $W_H(\underline{\varepsilon}) = W_H(\underline{0} - \underline{t}_{[0,\infty)}) = W_H(\underline{t}_{[0,\infty)}) < \infty$. მეორე მხრივ, $W_H(\underline{\delta}) = W_H(\underline{0} - \underline{i}_{[0,\infty)}) = W_H(\underline{i}_{[0,\infty)}) = \infty$; ასე რომ, არხში გადაცემული მიმდევრობის შეფასებისას დაშვებულ შეცდომათა სასრული რაოდენობა დეკოდირებისას საინფორმაციო სიმბოლოებში იწვევს შეცდომათა უსასრულო რაოდენობას. ამრიგად, საბოლოოდ ვრწმუნდებით ჩვენი მტკიცების სამართლიანობაში.

21. გისოსისებრი კოდების მაქსიმალური დამაჯერებლობის (ვიტერბის) დეკოდერი

შევისწავლოთ, თუ როგორ შეიძლება მოვახდინოთ დისკრეტულ უმეხსიერებო არხებში ნებისმიერი (N, R, L, T, M) გისოსისებრი კოდის მაქსიმალური დამაჯერებლობით დეკოდირება ეფექტური ალგორითმის გამოყენებით. ქვემოთ განხილული პროცედურა მიესადაგება ხვევად კოდებსაც, ვინაიდან ისინი წარმოადგენენ წრფივი გისოსისებრი კოდების კერძო შემთხვევას.

დავუშვათ, რომ $\underline{y}_{[0,L+T)} = [\underline{y}_0, \underline{y}_1, ..., \underline{y}_{L+T-1}]$ წარმოადგენს არხიდან მიღებულ მიმდევრობას, სადაც თითოეული $\underline{y}_u$ არის N ელემენტისგან შემდგარი არხის გამოსასვლელ სიმ-

ბოლოთა ბლოკი. ანალოგიურად, კოდური მიმდევრობა შეიძლება ჩაიწეროს როგორც $\underline{x}_{[0,L+T)}$ და იგი შედგება გისოსური დიაგრამის რომელიმე გზაზე მოთავსებულ არხის შესასვლელ სიმბოლოთა ერთობლიობისაგან. დავუშვათ, რომ ეს კოდური მიმდევრობა მიღებულია საინფორმაციო $\underline{i}_{[0,L+T)}$ მიმდევრობისაგან. მაქსიმალური დამაჯერებლობის დეკოდერი ირჩევს გადაცემულ საინფორმაციო მიმდევრობის ისეთ შემფასებელ $\hat{\underline{i}}_{[0,L+T)}$ მიმდევრობას (მიმდევრობებს), რომელიც ახდენს

$$P(\underline{y}_{[0,L+T)} \mid \underline{x}_{[0,L+T)}) = \prod_{u=0}^{L+T-1} P(\underline{y}_u \mid \underline{x}_u)$$

სიდიდის მაქსიმიზაციას, ანუ რაც იგივეა:

$$\log P(\underline{y}_{[0,L+T)} \mid \underline{x}_{[0,L+T)}) = \sum_{u=0}^{L+T-1} \log P(\underline{y}_u \mid \underline{x}_u) \qquad (21.1)$$

სტატისტიკის მაქსიმიზაციას. ვინაიდან მთელი ჩვენი შემდგომი მსჯელობისას ვთვლით, რომ $\underline{y}$ ფიქსირებულია, შეიძლება გამარტივებული სახით ჩავწეროთ:

$$\log P(\underline{y}_{[u,v]} \mid \underline{x}_{[u,v]}) = L_0(\underline{x}_{[u,v]}); \qquad (21.2)$$

თანაც ცხადია, რომ

$$L_0(\underline{x}_{[u,v]}) = \sum_{i=u}^{v} L_0(x_i). \qquad (21.3)$$

უნდა აღინიშნოს, რომ სინამდვილეში არ არის აუცილებელი L_0 ფუნქცია იყოს ლოგარითმული სახის. მაგრამ ამ შემთხვევაში ის წარმოადგენს სტატისტიკას, რომლის მაქსიმიზაციაც გვაძლევს მაქსიმალური დამაჯერებლობით

დეკოდირებას და, ამავე დროს, (21.3) ფორმულის შესაბამისად, ასეთი სტატისტიკა ადიტიურია, რაც ძალზე მოსახერხებელია პრაქტიკული რეალიზაციის თვალსაზრისით. მაგალითად, ორობითი სიმეტრიული არხისათვის შეგვიძლია მივიჩნიოთ:

$$L_0(\underline{x}_i) = -d_H(\underline{x}_i, \underline{y}_i)$$

ანუ $L_0(\underline{x}_i)$ ტოლია x_i -ის გადაცემისას არხში მომხდარი „შეცდომების“ რიცხვისა მინუს ნიშნით.

(N, R, L, T, M) გისოსისებრი კოდების მაქსიმალური დამაჯერებლობით დეკოდირება დაფუძნებულია ერთ მარტივ პრინციპზე, რომელიც შეიძლება მოყვანილ იქნას შემდეგი ლემის სახით.

ლემა 21.1 (არაოპტიმალურობის პრინციპი). თუ $\underline{i}'_{[0,u]}$ და $\underline{i}''_{[0,u]}$ გზები ბოლოვდებიან გისოსის ერთსა და იმავე კვანძში და

$$L_0(\underline{x}'_{[0,u]}) > L_0(\underline{x}''_{[0,u]}) \qquad (21.4)$$

მაშინ $\underline{i}''_{[0,u]}$ არ შეიძლება იყოს იმ $\underline{i}_{[0,L+T)}$ გზის (გზების) პირველი $u+1$ შტო, რომელიც (რომლებიც) ახდენს $L_0(\underline{x}_{[0,L+T)})$ სიდიდის მაქსიმიზაციას.

დამტკიცება. დავუშვათ საწინააღმდეგო პირობა, რომ $\underline{i}''_{[0,L+T)} = \underline{i}''_{[0,u]} * \underline{i}''_{(u,L+T)}$ მიმდევრობა ახდენს L_0 სტატისტიკის მაქსიმიზაციას. (აქ და შემდგომში სიმბოლო $*$ აღნიშნავს ორი მიმდევრობის შეერთებას). განვიხილოთ მიმდევრობა (გზა): $\underline{i}'_{[0,L+T)} = \underline{i}'_{[0,u]} * \underline{i}''_{(u,L+T)}$. კოდირებული $\underline{x}'_{[0,L+T)} = \underline{x}'_{[0,u)} * x_{[u,L+T)}$

მიმდევრობა უნდა შეიცავდეს $\underline{x}^{|}_{[0,L+T)} = \underline{x}^{||}_{[0,L+T)}$ სეგმენტს, ვინაიდან $\underline{i}^{|}_{[0,u]}$ და $\underline{i}^{||}_{[0,u]}$ ბოლოვდებიან ერთსა და იმავე კვანძში და ამის შედეგად ორივე შემთხვევაში ჩვენ ვიყენებთ ერთსა და იმავე საინფორმაციო $\underline{i}^{||}_{(u,L+T)}$ მიმდევრობას. მაშასადამე, თუ გამოვიყენებთ (21.4) ფორმულას, მივიღებთ:

$$L_0(\underline{x}^{|}_{[0,L+T)}) = L_0(\underline{x}^{|}_{[0,u]} * \underline{x}^{||}_{[u,L+T)}) =$$
$$= L_0(\underline{x}^{|}_{[0,u]}) + L_0(\underline{x}^{||}_{[u,L+T)}) > L_0(\underline{x}^{||}_{[0,u]}) + L_0(\underline{x}^{||}_{[u,L+T)}).$$

ამრიგად, გვაქვს:

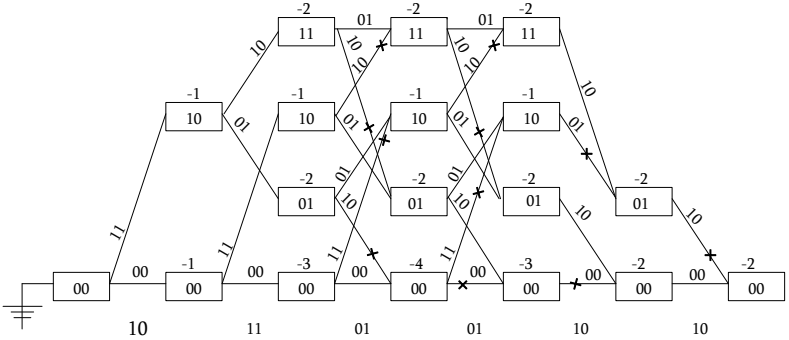
$$L_0(\underline{x}^{|}_{[0,L+T)}) > L_0(\underline{x}^{||}_{[0,L+T)}),$$

რაც ეწინააღმდეგება ჩვენს დაშვებას იმის შესახებ, რომ $\underline{i}^{||}_{[0,L+T)}$ ახდენს L_0 სტატისტიკის მაქსიმიზაციას. □

ახლა, მაგალითის საფუძველზე ვუჩვენოთ, თუ როგორ შეიძლება იქნას გამოყენებული არაოპტიმალურობის პრინციპი ეფექტური მაქსიმალური დამაჯერებლობის დეკოდირების პროცედურის შესამუშავებლად გისოსისებრი კოდებისათვის. მე-19 ნახაზზე ნაჩვენებია გისოსისებრი მუდმივი ხვევადი ორობითი ($N = 2,\ R = 1/2,\ L = 4,\ T = 2,\ M = 2$) კოდი, რომელიც გენერირდება 14ა ნახაზზე მოცემულ კოდერით. დავუშვათ, რომ ამ კოდის ერთ-ერთი კომბინაციის გადაცემის შედეგად არხში არსებული ხელშეშლების გამო (გვაქვს ორობითი სიმეტრიული არხი გადასასვლელი ალბათობით $\varepsilon < 1/2$) დეკოდერის შესასვლელზე მიეწოდება მიმდევრობა - $\underline{y}_{[0,6)} = [1,0,1,1,0,1,0,1,1,0,0,0]$. დეკოდირების ზემოთ მოყვანილი L_0 სტატისტიკის საზომად გამოვიყენოთ

მიღებულ მიმდევრობასა და კოდირებულ მიმდევრობას შორის შეცდომათა რიცხვი მინუს ნიშნით. ციფრი, რომელიც მიწერილია თითოეულ კვანძზე ანუ „კოდერის მდგომარეობისათვის“, წარმოადგენს ამ კვანძში მოსული საუკეთესო გზის სტატისტიკას. გისოსის 1 და 2 სიღრმეებზე მოთავსებული კვანძებისათვის არსებობს მათკენ მიმავალი მხოლოდ თითო გზა და ამიტომ ისინი წარმოადგენენ საუკეთესო გზებს. საინტერესო მოვლენა იწყება სიღრმეზე 3. ჩვენ ვხედავთ, რომ იმ ორი გზიდან, რომლებიც შედის, მაგალითად, $[1,1]$ მდგომარეობაში, ზედა მათგანისათვის $L_0 = -2$ ხოლო ქვედა მათგანისათვის $L_0 = -3$. გისოსზე საუკეთესო გზის მოძებნასთან დაკავშირებული შემდგომი გამოკვლევებიდან ჩვენ, არაოპტიმალურობის პრინციპიდან გამომდინარე, ვაგდებთ (უგულებელვყოფთ) ქვედა გზას და ვტოვებთ მხოლოდ ზედა მათგანს. მე-19 ნახაზზე გადაგდებულ გზებზე მოთავსებულია ჯვრის გამოსახულება. მაშასადამე, $[1,1]$ მდგომარეობაში დარჩენილია მხოლოდ ზედა გზა, ხოლო ქვედა გზა გადაგდებულია, ვინაიდან მასზე დასმულია "$\times$" ნიშანი. ამის შემდეგ ჩვენ ვიმეორებთ იგივე პროცედურას დარჩენილი სამი მდგომარეობისათვის სიღრმეზე 3 და თითოეულ მდგომარეობას აღვნიშნავთ (ზემოდან ვაწერთ) დატოვებულ საუკეთესო გზის L_0 მნიშვნელობით. ვინაიდან აღწერილი პროცედურის სრულად განხორციელების შემდეგ სიღრმეზე 3 თითოეულ კვანძში დატოვებულია თითო გზა, ამიტომ თავდაპირველად ზუსტად ორი გზა იქნება შემავალი თითოეულ მდგომარეობაში სიღრმეზე 4. მდგომარეობებს სიღრმეზე 4 ჩვენ ვამუშავებთ იმავე წესით, როგორც

სიღრმეზე 3 და ვაგდებთ თითოეულ მდგომარეობაში მიმავალ უარეს გზებს. ამრიგად, სიღრმეზე 4 (ისევე როგორც სიღრმეზე 3) საბოლოოდ რჩება ოთხი საუკეთესო გზა (თითო ყველა ოთხ კვანძში) და, აქედან გამომდინარე, სიღრმეზე 5 თავდაპირველად თითოეულ კვანძში ისევ შედის ზუსტად ორი გზა. უნდა აღინიშნოს, რომ სიღრმეზე 5 ჩვენ უკვე შეგვიმცირდა კვანძების რაოდენობა ოთხიდან ორამდე. მდგომარეობებს ამ სიღრმეზე ჩვენ ვამუშავებთ იმავე წესით, როგორც 3 და 4 სიღრმეებზე და ვტოვებთ ორ საუკეთესო გზას - თითოს ყველა მდგომარეობაში. მაშასადამე, სიღრმეზე 6 დარჩენილ ერთადერთ $[0,0]$ მდგომარეობაში გვაქვს ორი შემავალი გზა. ჩვენ ვამუშავებთ ამ უკანასკნელ მდგომარეობაში შემავალ გზებსაც და ვაგდებთ უარეს მათგანს. ამ მომენტისათვის აღმოჩნდება, რომ გისოსზე დარჩენილია ერთადერთი (საუკეთესო) $\underline{x}_{[0,6)} = [0,0,1,1,0,1,1,1,0,0,0,0]$ გზა, რომელიც შეესაბამება საინფორმაციო მიმდევრობას - $\underline{i}_{[0,6)} = [0,1,0,0,0,0]$. ვინაიდან ალგორითმის ფუნქციონირების განმავლობაში ჩვენ მუდამ ვიყავით არაოპტიმალურობის პრინციპის ერთგული, ე. ი. ყოველთვის ვაგდებდით ისეთ გზებს, რომლებიც არ შეიძლება ყოფილიყვნენ ოპტიმალურები, შეგვიძლია დავასკვნათ, რომ სწორედ დარჩენილი გზა წარმოადგენს ოპტიმალურ გზას. ამრიგად, მაქსიმალური დამაჯერებლობის დეკოდერით მიღებული გადაწყვეტილება შეესაბამება გზას - $\hat{\underline{i}}_{[0,L)} = \hat{\underline{i}}_{[0,4)} = [0,1,0,0]$.



ნახ. 19. ორობითი სიმეტრიულ არხში მაქსიმალური დამაჯერებლობით დეკოდირების მაგალითი

განხილული მაგალითის ბაზაზე შეიძლება შემუშავებულ იქნას ყველა სახის დისკრეტულ უმეხსიერებო არხში ნებისმიერი (N, R, L, T, M) გისოსისებრი კოდის მაქსიმალური დამაჯერებლობით დეკოდირების ზოგადი მეთოდი. M სიღრმემდე მხოლოდ თითო გზა შედის ყველა მდგომარეობაში, რომელთაც მიეწერებათ შესაბამისი L_0 სტატისტიკები. $M+1$ სიღრმეზე გვაქვს თითოეულ მდგომარეობაში შემავალი ზუსტად $m = 2^{NR}$ გზა. ჩვენ ვაგდებთ საუკეთესო გზის გარდა ყველა დანარჩენ გზას და დატოვებული გზის სტატისტიკას ვანიჭებთ შესაბამის მდგომარეობას. ამრიგად, $M+2$ სიღრმეზე თითოეულ მდგომარეობაში შედის ისევ ზუსტად m გზა და ეს მდგომარეობები მუშავდება იგივე პრინციპით, როგორც $M+1$ სიღრმეზე. გისოსის საბოლოო $L+T$ სიღრმის მიღწევისა და ამ ეტაპზე მდგომარეობების დამუშავების შემდეგ ჩვენ გვრჩება თითო გზა ყველა m^{M-T}

საბოლოო მდგომარეობაში. ამ დარჩენილი გზებიდან, რომლებიც მოიცავენ მთელი გისოსის სიგრძეს, ჩვენ ვირჩევთ უდიდესი L_0 სტატისტიკის მქონეს და სწორედ ის იქნება ოპტიმალური გზა, ე. ი. მაქსიმალური დამაჯერებლობის დეკოდერით არჩეული გზა. უნდა აღინიშნოს, რომ, ჩვეულებრივ გისოსისებრი კოდისთვის $M = T$ და ამ ბოლო გადარჩევის ჩატარება საჭიროებას აღარ წარმოადგენს.

გისოსისებრი კოდების მაქსიმალური დამაჯერებლობით დეკოდირების აღწერილი მეთოდი შემუშავებული იყო ა. ვიტერბის მიერ და ამიტომ მას უწოდებენ „ვიტერბის ალგორითმს“ [20]. ა. ვიტერბიმ დეკოდირების ეს პროცედურა გამოიყენა როგორც თეორიული ინსტრუმენტი დროში ცვალებადი ხვევადი კოდებისათვის (13.4) ფორმულით მოცემული საზღვრის დასამტკიცებლად (უნდა აღინიშნოს, რომ მან მიიღო საზღვარი უფრო მაღალი სიჩქარეებისთვისაც $R_0 \le R < C$). იმ დროისთვის ა. ვიტერბიმ არ იცოდა, რომ აღწერილი პროცედურა წარმოადგენდა მაქსიმალური დამაჯერებლობით დეკოდირებას. პირველად ეს შენიშნა ჯ. ომურამ [21], რომელმაც უჩვენა, რომ სინამდვილეში ვიტერბის ალგორითმი არის ოპტიმალური დეკოდირების პრობლემის „დინამიკური პროგრამირების“ მეთოდებით გადაჭრა. ვიტერბის ალგორითმის თვალნათლივ წარმოდგენისა და ეფექტური ანალიზისათვის დიდი როლი შეასრულა დ. ფორნის მიერ ხვევადი კოდების აღწერამ „გისოსისებრი დიაგრამის“ გამოყენებით [11].

პარაგრაფის დასასრულს ორიოდე სიტყვით შევჩერდეთ ვიტერბის დეკოდერის პრაქტიკული რეალიზაციის საკი-

თხზე. აღწერილი ალგორითმიდან ჩანს, რომ მისი „სირთულე“ (ანუ ოპერაციების რაოდენობა რომელიმე უნივერსალურ ეგმ-ზე) იზრდება ექსპონენციალურად მეხსიერების (M) ზრდასთან ერთად. დღეისათვის პრაქტიკაში გამოყენებული დეკოდერებისათვის მეხსიერება $M = 9$ და $M = 10$ წარმოადგენს ზღვრულ სიდიდეს. მაგრამ ასეთი მეხსიერების მქონე $R = 1/2$ სიჩქარიანი მუდმივი ხვევადი კოდერები, კოდური შემზღუდველი სიგრძის არც თუ ისე დიდი მნიშვნელობების $N_t = (M+1)N = 18$ (ან 20) მიუხედავად, მათი ვიტერბის ალგორითმით დეკოდირებისას გვაძლევენ ძალზე ეფექტურ მაკორექტირებელ მახასიათებლებს შედარებით „ცუდ“ კავშირის არხებში, ე. ი. სიგნალ/ხელშეშლის ფარდობის დაბალი მნიშვნელობებისათვის. ასეთი კოდები და მათი დეკოდირების ვიტერბის ალგორითმი ფართოდ გამოიყენება შიგა საფეხურზე კასკადური კოდირების სისტემებში [22].

22. ხისმაგვარი კოდების დეკოდირება - ფანოს მეტრიკა

ახლა შევისწავლოთ ასეთი საკითხი - რომელი მეთოდითაა ყველაზე უფრო მიზანშეწონილი ხისმაგვარი კოდების დეკოდირება. ჩვენ, რა თქმა უნდა, შეგვიძლია განვიხილოთ მუდმივი ხვევადი კოდერები, როგორც ხისმაგვარი, ასევე გისოსისებრი კოდების წარმომქმნელებად და ამიტომ, ცხადია, ქვემოთ მოყვანილი შედეგები მიესადაგება ხვევად კოდებსაც. ამ პარაგრაფში ჩვენ შევისწავლით ისეთ მუდმივ ხვევად კოდერებს, რომელთა მეხსიერებაც - M ძალზე დიდია, ამიტომ

განვიხილავთ სწორედ ხისმაგვარ კოდებს, ვინაიდან უაზრობაა საუბარი მათი გისოსისებრი ან მდგომარეობათა დიაგრამით აღწერაზე. ამასთან, ძალზე დიდი „სირთულის“ გამო შეუძლებელია მათი დეკოდირებისას ვიტერბის ალგორითმის გამოყენებაც.

როდესაც L საკმაოდ დიდია, გზათა რიცხვი 2^{NRL} ხისმაგვარ დიაგრამაში აღწევს ისეთ სიდიდეს, რომ წარმოუდგენელია შევადაროთ მიღებული მიმდევრობა თითოეულ გზას დიაგრამით მოცემული კოდის მაქსიმალური დამაჯერებლობით დეკოდირების შესასრულებლად. მაგალითად, როდესაც $R = 1/N$ და $L = 100$ (რაც საკმაოდ „მცირე“ რიცხვია, თუ მხედველობაში მივიღებთ დღეისათვის პრაქტიკაში გამოყენებულ ხისმაგვარ კოდებს), მაშინ დიაგრამაზე არსებობს $2^{100} \approx 10^{30}$ განსხვავებული გზა. მაგრამ, ვინაიდან ხეზე მოთავსებული ადრეული კვანძებიდან გამოდის გზების ძალზე დიდი რაოდენობა, შეგვიძლია ვივარაუდოთ, რომ დეკოდირების ხარისხი მნიშვნელოვნად არ გაუარესდება მაქსიმალური დამაჯერებლობით დეკოდირებასთან შედარებით თუ ჩვენ შემდგომი განხილვისას უგულებელვყოფთ ისეთი კვანძებიდან გამომავალ ყველა გზას, რომლებისთვისაც მათში მიმავალი საუკეთესო გზის მეტრიკა ძალზე ცუდია. უნდა აღინიშნოს, რომ სწორედ ეს არის ხისმაგვარი კოდების გამოყენების არსი. „ცუდი“ გზების უგულვებელსაყოფად ჩვენ გვჭირდება მათი ხარისხის რაღაც აბსოლუტური ზომა, რომელიც მხედველობაში მიიღებს იმ ფაქტსაც, რომ გადასაგდებ გზებს აქვთ სხვადასხვა სიგრძეები. ასეთი ზომის ან „ხარისხის მეტრიკის“ განსაზღვრისას ჩვენ, ბუნებრივია, გადავდი-

ვართ ისეთი კოდების დეკოდირების პრობლემის შესწავლაზე, რომლებშიც კოდურ სიტყვებს (ჩვეულებრივი ბლოკური კოდებისაგან განსხვავებით) აქვთ სხვადასხვა სიგრძეები.

დავუშვათ, რომ $(\underline{x}_1, \underline{x}_2, ..., \underline{x}_S)$ S სიტყვის შემცველი კოდია, სადაც:

$$\underline{x}_s = (x_{s1}, x_{s2}, ..., x_{sN_s}), \quad s = 1, 2, ..., S,$$

კოდური სიტყვებია, რომელთა $N_1, N_2, ..., N_s$ სიგრძეებიც ზოგად შემთხვევაში განსხვავებულია. დავუშვათ, რომ დისკრეტულ უმეხსიერებო არხში s შეტყობინების გადაცემას შეესაბამება $\underline{x}_s$ სიტყვის გადაცემა. დეკოდერმა უნდა მოახდინოს არხში გადაცემული შეტყობინების შეფასება - $\hat{s}$. აღვნიშნოთ:

$$N = \max(N_1, N_2, ..., N_s)$$

და დეკოდერისგან მოვითხოვოთ შეაფასოს s შეტყობინება მიღებული $\underline{y} = [y_1, y_2, ..., y_N]$ მიმდევრობის ბაზაზე. დისკრეტული უმეხსიერებო არხით გადაცემული s შეტყობინების სიგრძის შესახებ ნებისმიერი ინფორმაციის თავიდან აცილების მიზნით პირობით დავუშვათ, რომ თითქოს $\underline{x}_s$ სიტყვის გადაცემის შემდეგ (თუ $N_s < N$) არხით გადაიცა $N - N_s$ სიმბოლოსგან შედგენილი კუდი. ეს კუდი მიღებულია არხის შესასვლელი ალფაბეტის სიმბოლოთა ერთმანეთისაგან დამოუკიდებლად არჩევით, $Q(x)$ ალბათობათა განაწილების ფუნქციის შესაბამისად. იგი მიერთებულია პირითად $\underline{x}_s$ სიტყვასთან და არხის გამოსასვლელზე მოთავსებულმა დეკოდერმა იცის მხოლოდ $Q(x)$ განაწილება, რომელიც გამოყენებულია მისი სიმბოლოების არჩევისას. დავუშვათ, რომ

შეტყობინებები არ არის თანაბარალბათური და P_s სიდიდით აღვნიშნოთ s შეტყობინების გადაცემის ალბათობა.

ახლა გადავიდეთ დეკოდირების ისეთი წესის მოძებნაზე, რომელიც მოახდენს შეცდომითი დეკოდირების ალბათობის მინიმიზაციას. აღვნიშნოთ $P(s, \underline{y})$ სიდიდით ისეთი ხდომილების ალბათობა, როდესაც არხით გადაცემულ s შეტყობინებას შეესაბამება მიღებული $\underline{y}$ კოდური სიტყვა. ვინაიდან განვიხილავთ უმეხსიერებო არხს და ის შემთხვევითი სიმბოლოები, რომლებიც $\underline{x}_s$-ს მოსდევენ კუდის სახით, დამოუკიდებლად არის არჩეული, გვაქვს:

$$P(s, \underline{y}) = P_s \prod_{n=1}^{N_s} P(y_n \mid x_{sn}) \prod_{t=N_s+1}^{N} P_Q(y_t) . \qquad (22.1)$$

აქ $P_Q(y)$ წარმოადგენს y-ის მიღების ალბათობას თუ არხში გაგზავნილია შემთხვევითი სიმბოლო, ე. ი.

$$P_Q(y) = \sum_{x \in A} P(y \mid x) Q(x) , \qquad (22.2)$$

სადაც A არხის შესასვლელი ალფაბეტია. შეცდომითი დეკოდირების ალბათობა მინიმიზირდება იმ შემთხვევაში, თუ s-ის მნიშვნელობად არჩეულია ისეთი $\hat{s}$ სიდიდე, რომელიც ახდენს (22.1) ფორმულით მოცემული $P(s, \underline{y})$ გამოსახულების მაქსიმიზაციას, ანუ რაც იგივეა, ახდენს შემდეგი გამოსახულების მაქსიმიზაციას:

$$\frac{P(s, \underline{y})}{\prod_{i=1}^{N} P_Q(y_i)} = P_s \prod_{n=1}^{N_s} \frac{P(y_n \mid x_{sn})}{P_Q(y_n)} . \qquad (22.3)$$

აქ $P(s, \underline{y})$ ჩვენ უბრალოდ გავყავით დადებით მუდმივ სიდიდეზე, რომელიც არ არის დამოკიდებული s-ზე. ახლა თუ ავიღებთ (22.3) გამოსახულების ლოგარითმს, დავრწმუნდებით, რომ შეცდომითი დეკოდირების ალბათობის მინიმიზაცია ხდება s-ის მნიშვნელობად ისეთი $\hat{s}$-ის არჩევისას, რომელიც ახდენს შემდეგი სტატისტიკის მაქსიმიზაციას:

$$L_F(s, \underline{y}) = \sum_{n=1}^{N_s} [\log \frac{P(y_n \mid x_{sn})}{P_Q(y_n)} - \frac{1}{N_s} \log \frac{1}{P_s}]. \qquad (22.4)$$

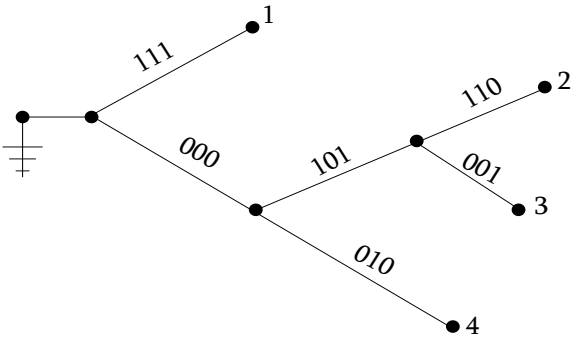
ამრიგად, ადგილი აქვს ერთ საოცარ, მაგრამ სასიამოვნო ფაქტს: „დეკოდირების მეტრიკა“ s შეტყობინებისათვის დამოკიდებულია $\underline{y}$ სიტყვის მხოლოდ იმ ნაწილზე, რომელსაც აქვს ისეთივე სიგრძე, როგორიც s-ს.

ახლა შევისწავლოთ, თუ როგორ შეიძლება (22.4) ფორმულით მოცემული მეტრიკის გამოყენება ხისმაგვარი კოდების „თითქმის მაქსიმალური დამაჯერებლობით დეკოდირებისას“ ისე, რომ საჭირო არ გახდეს მთელი ხისმაგვარი დიაგრამის გამოკვლევა. მაგალითისათვის განვიხილოთ ოროობითი შესასვლელიანი არხი და $R = 1/N$ სიჩქარის მქონე ხისმაგვარი კოდი, როდესაც $N = 3$. დავუშვათ, რომ ჩვენ ნაწილობრივ უკვე გამოვიკვლიეთ ხე, როგორც ესაა ნაჩვენები მე-20 ნახაზზე. ბუნებრივია, იბადება კითხვა: რომელი მათგანი უნდა განვაგრძოთ ოთხ საბოლოო კვანძისაგან, რომ შევძლოთ „დროის უმეტეს მონაკვეთზე“ მაინც ჩავატაროთ იმავე კოდირებული გზის გამოკვლევა, რომელსაც დაამუშავებდა მაქსიმალური დამაჯერებლობის დეკოდერი? გავიხსენოთ, რომ მაქსიმალური დამაჯერებლობის დეკოდერი ეკვივალენტუ-

რია ისეთი დეკოდერისა, რომელიც ახდენს თანაბარალბათური კოდური სიტყვებისათვის შეცდომის ალბათობის მინიმიზაციას. მაგრამ, თუ 2^{NRL} კოდურ სიტყვას ხისმაგვარ კოდში მივიჩნევთ თანაბარალბათურად, ეს ნიშნავს, რომ ნებისმიერ კვანძში, სადაც ხდება განშტოება კოდერი თანაბარი ალბათობით მიჰყვება ერთ-ერთს 2^{NR} შტოთაგან. აქედან გამომდინარე, მე-20 ნახაზზე მოცემულ შეტყობინებებს აქვთ შემდეგი ალბათობები: $P_1 = 2^{-1}$, $P_2 = P_3 = 2^{-3}$ და $P_4 = 2^{-2}$. ამის შემდეგ უკვე შესაძლებელია ამ ალბათობების (22.4) ფორმულაში გამოყენება თითოეული შეტყობინებისათვის დეკოდირების მეტრიკის საპოვნელად. ამრიგად, თუ ჩვენ დროის ყველა მომენტში გავაგრძელებთ კვანძს უდიდესი მეტრიკით და ამ პროცედურას ჩავატარებთ მთელი ხის საბოლოო კვანძის მიღწევამდე, შეგვიძლია თითქმის დარწმუნებული ვიყოთ იმაში, რომ გამარჯვებული დარჩება იგივე გზა, რომელსაც იპოვიდა მაქსიმალური დამაჯერებლობის დეკოდერი უზომოდ დიდი გადარჩევის შედეგად. ეს არის სწორედ ის ძირითადი იდეა, რომელიც ჩადებულია მიმდევრობით დეკოდირებაში და რომლის ორ ძირითად ვარიანტს აღვწერთ და შევისწავლით მომდევნო ორ პარაგრაფში.

ახლა, სანამ გადავიდოდეთ მიმდევრობითი დეკოდირების დეტალურად შესწავლაზე, კიდევ ერთხელ ყურადღებით დავაკვირდეთ (22.4) ფორმულით მოცემულ მეტრიკას ნაწილობრივ გამოკვლეული ხისმაგვარი კოდისათვის. თუ s კვანძი მოთავსებულია ხის ფესვიდან u სიღრმეზე, მაშინ $N_s = N_u$ (სადაც N არის ერთ შტოზე მოსული არხის სიმბო-

ლოთა რიცხვი) და ალბათობა იმისა, რომ კოდერი მიაღწევს ხის ამ კვანძს, ტოლია $P_s = 2^{-NRu}$ -ის.



ნახ. 20. ორობითი შესასვლელიან არხში მომუშავე $R = 1/3$ სიჩქარის მქონე ნაწილობრივ გამოკვლეულიხისმაგვარი კოდი

თუ ჩავსვამთ ამ მნიშვნელობებს (22.4) ფორმულაში, მივიღებთ:

$$L_F(s, \underline{y}) = \sum_{n=1}^{N_s} [\log \frac{P(y_n \mid x_{sn})}{P_Q(y_n)} - R], \qquad (22.5)$$

რაც წარმოადგენს საბოლოო და ძალზე მოსახერხებელ მეტრიკას ხისმაგვარი კოდებისათვის.

(22.5) ფორმულით მოცემული მეტრიკა პირველად გამოყენებული იყო რ. ფანოს მიერ [23] (სწორედ ამიტომ ვხმარობთ F ინდექსს). უნდა აღინიშნოს რ. ფანოს საოცარი ინტუიცია, ვინაიდან აღნიშნული მეტრიკის პოსტულირება მან მოახდინა ყოველგვარი მათემატიკური დასაბუთების გარეშე,

გამოიყენა რა იგი მიმდევრობითი დეკოდირების ახალი ეფექტური ალგორითმის აღსაწერად. საკითხის ზემოთ მოყვანილი ანალიზური დადასტურება მიღებული იყო თითქმის ათი წლის შემდეგ ჯ. მესის მიერ [24].

ორიოდე სიტყვით შევჩერდეთ ალბათობათა $Q(x)$ განაწილების შესახებ, რომელიც გამოყენებული უნდა იქნას (22.2) ფორმულაში. ქვემოთ მოყვანილი მარტივი მსჯელობის საფუძველზე ვრწმუნდებით, რომ იგი უნდა იყოს (2.2) ფორმულით მოცემული გამოსახულების მამინიზებელი განაწილება. თუ ჩვენი ხისმაგვარი კოდი მაქსიმალური დამაჯერებლობით დეკოდირებისას გვაძლევს შეცდომის ალბათობას, რომელიც შემოსაზღვრულია ისევე, როგორც (22.5) ფორმულაში, მაშინ მას უნდა ჰქონდეს ისეთი ხისმაგვარი კოდის ხასიათი, სადაც სიმბოლოები არჩეულია ერთმანეთისაგან დამოუკიდებლად (2.2) ფორმულით მოცემული $Q(x)$ -ის შესაბამისად. თუ განვიხილავთ ნებისმიერ ნაწილობრივ გამოკვლეულ უბანს, ჩვენ შეგვიძლია დავუშვათ, რომ მომდევნო სიმბოლოებს ჯერ გამოუკვლევ უბანზე აქვთ ისეთი ხასიათი, თითქოს ისინი არჩეულნი არიან ერთმანეთისაგან დამოუკიდებლად $Q(x)$ -ის შესაბამისად და ეს არის ერთადერთი აპრიორული ინფორმაცია, რომელიც ჩვენ წინასწარ გვჭირდება ვიცოდეთ (ვივარაუდოთ) ამ სიმბოლოების შესახებ, სანამ არ გავაგრძელებთ კვლევის პროცესს ხისმაგვარ დიაგრამაზე.

რიცხვითი მაგალითი მკითხველს საშუალებას მისცემს უკეთ „შეიგრძნოს“ ფანოს მეტრიკა. განვიხილოთ ხისმაგვარი კოდი, განკუთვნილი ორობითი სიმეტრიული არხისათვის გადასასვლელი ალბათობით $\varepsilon = 0.045$. ნებისმიერი

არხისათვის ორობითი შესასვლელით, და მათ შორის, ორობითი სიმეტრიული არხისათვისაც, მამინიზებელი განაწილება (2.2) ფორმულაში არის $Q(0) = Q(1) = 1/2$. თუ $\varepsilon = 0.045$ (2.2) ფორმულიდან ვპოულობთ $R_0 = 0.5$ და ბუნებრივია, რომ მოცემულ მაგალითში ვირჩევთ ხისმაგვარს კოდს სიჩქარით $R = 1/2$. (22.5) ფორმულიდან გამომდინარეობს, რომ ხისმაგვარი დიაგრამის s-ური გზის n-ური სიმბოლოს ფანოს მეტრიკა ტოლია:

$$L_F(x_{sn}) = \log \frac{1-0.045}{0.5} - 0.5 \approx +0.50 \quad \text{თუ } x_{sn} = y_n$$

და

$$L_F(x_{sn}) = \log \frac{1-0.045}{0.5} - 0.5 \approx -3.50 \quad \text{თუ } x_{sn} \neq y_n \ .$$

პრაქტიკაში ჩვეულებრივ მეტრიკის მასშტაბს ცვლიან ისე, რომ მეტრიკის ყველა მნიშვნელობა რაც შეიძლება ახლოს იყოს რომელიმე მთელ რიცხვთან და ამით საგრძნობლად მარტივდება დეკოდირების პროცესი. ჩვენს მაგალითში გავამრავლოთ ორივე მეტრიკა 2-ზე; მივიღებთ:

$$L_F(x_{sn}) = \begin{cases} +1 & \text{თუ} \quad x_{sn} = y_n \\ -7 & \text{თუ} \quad x_{sn} \neq y_n \end{cases} .$$

მოცემულ მაგალითში ფანოს მეტრიკა ჩაწერილია გამარტივებული სახით. ეს მოხდა ორი მიზნით: პირველი - რათა ხაზი გაგვესვა იმ ფაქტისათვის, რომ დეკოდირების კონკრეტულ სიტუაციაში, მიღებული $\underline{y}$ ვექტორი შეიძლება განხილულ იქნას როგორც ფიქსირებული ვექტორი და ამიტომ ამოღებულ იქნას მეტრიკის გამოსახულებიდან; მეორე - აღგვენიშნა ის ფაქტიც, რომ ფანოს მეტრიკა წარმოადგენს

სიმბოლოების ადიტიურ ფუნქციას კოდირებული $\underline{x}$ გზის გასწვრივ. ჩვენ შემდგომშიც გამოვიყენებთ ამ გამარტივებულ აღნიშვნებს და ჩავწერთ:

$$L_F(\underline{x}_s) = \sum_{n=1}^{N_s} L_F(x_{sn}), \tag{22.6}$$

სადაც

$$L_F(x_{sn}) = \log \frac{P(y_n \mid x_{sn})}{P_Q(y_n)} - R \ . \tag{22.7}$$

23. მიმდევრობითი დეკოდირება - სტეკ-ალგორითმი

მიმდევრობითი დეკოდირება არის ხისმაგვარი კოდის ნებისმიერი ისეთი დეკოდირების პროცედურის ზოგადი სახელწოდება, რომელიც არხიდან მიღებული $\underline{y}_{[0,L+T)}$ მიმდევრობის საფუძველზე, ხისმაგვარი კოდირებული დიაგრამის თანდათანობით (მიმდევრობითი) გამოკვლევით ირჩევს ყველაზე უფრო მეტად სარწმუნო გადაცემულ გზას. ამასთან, გამოკვლევის ხასიათზე წაყენებულია შემდეგი მოთხოვნები:

(ა) ყოველი ახლად გამოსაკვლევი კვანძი უნდა იმყოფებოდეს უკვე დამუშავებული კვანძის მიმართ ხის ფესვიდან მომდევნო სიღრმეზე;

(ბ) ხისმაგვარი კოდების ანსამბლის (რომელშიც გამოყენებული კოდი „ტიპური“ წევრია) მახასიათებელი სტატისტიკური $Q(x)$ განაწილების გარდა, დეკოდერისათვის არავი-

თარი სხვა ინფორმაცია არ არის ხელმისაწვდომი ხის ჯერ კიდევ გამოუკვლევი ნაწილის შესახებ.

წინა პარაგრაფში მოყვანილ მსჯელობას მკითხველი უეჭველად მიჰყავს იმ დასკვნამდე, რომ ყველაზე უფრო „ბუნებრივი“ დეკოდირების ალგორითმი მეხსიერებაში ინახავს უკვე გამოკვლეულ ყველა კვანძამდე არსებული გზების ფანოს მეტრიკას და შემდეგ ხის ფესვიდან მომდევნო სიღრმემდე განაგრძობს გზას უდიდესი ფანოს მეტრიკით. მოცემული მარტივი პროცედურა წარმოადგენს პრაქტიკაში ფართოდ გავრცელებულ სტეკ-ალგორითმის ბაზას, რომელიც პირველად განხილული იყო კ. ზიგანგიროვის მიერ [25], ხოლო შემდეგ მისგან დამოუკიდებლად შეიმუშავა და გაანალიზა ფ. ჯელინეკმა [26]. სახელი „სტეკ“ (stack - გროვა, დასტა) ასახავს დეკოდერის ფუნქციონირების წესს: იგი ინახავს უკვე შესწავლილი კვანძების შესაბამისი მეტრიკების მნიშვნელობებს სტეკში (ისე, რომ კვანძები სორტირებული არიან ზემოდან ქვემოთ მეტრიკების კლებადობის მიხედვით) და თითოეულ მომდევნო ბიჯზე განაგრძობს კვანძს, მოთავსებულს სტეკის თავზე.

$R = 1/N$ სიჩქარის მქონე ხისმაგვარი კოდებისათვის, რომელთათვისაც ხის თითოეული კვანძიდან გამოდის მხოლოდ ორი შტო, სტეკ-ალგორითმის მუშაობის პრინციპი ყველაზე უფრო ზოგადი სახით შეიძლება ჩამოყალიბდეს შემდეგნაირად: აღვნიშნოთ $\underline{x}_0$-ით და $\underline{x}_1$-ით კოდირებული შტოები, რომლებიც მიიღება საინფორმაციო სიმბოლოების, შესაბამისად, 0-ისა და 1-ის მიწოდებით რომელიმე i-ური გზის გაგრძელებისას. თითოეული ჩანაწერი სტეკში შეიცავს

$[\underline{i}, L_F(\underline{x})]$ წყვილს, სადაც i გზის ნომერია ხისმაგვარ დიაგრამაში, $\underline{x}$ - ამ გზაზე მოთავსებული კოდირებული სიმბოლოებია, ხოლო $L_F(\underline{x})$ $\underline{x}$-ის ფანოს მეტრიკაა. აღვნიშნოთ $\wedge$ სიმბოლოთი „ცარიელი მწკრივი“, რომელიც წარმოადგენს თავდაპირველ გზას ხის ფესვისკენ. როდესაც $\underline{i} = \wedge$, გვაქვს $\underline{x} = \wedge$ და $L_F(\underline{x}) = 0$.

ალგორითმი შედგება შემდეგი ბიჯებისაგან:

ბიჯი 0: გავასუფთაოთ სტეკი და შემდეგ შევიტანოთ ჩანაწერი $[\wedge, 0]$.

ბიჯი 1: გავაგრძელოთ სტეკის თავზე მდებარე ჩანაწერის $[\underline{i}, L_F(\underline{x})]$ შესაბამისი გზა და მოვახდინოთ $[\underline{i} * 0, L_F(\underline{x}) + L_F(\underline{x}_0)]$ და $[\underline{i} * 1, L_F(\underline{x}) + L_F(\underline{x}_0)]$ სიდიდეების ფორმირება. ამოვიღოთ ჩანაწერი $[\underline{i}, L_F(\underline{x})]$ სტეკიდან.

ბიჯი 2: შევიტანოთ ახლად ფორმირებული ორი ჩანაწერი სტეკში ისე, რომ სტეკი დარჩეს სორტირებული ზემოდან ქვემოთ მეტრიკების კლებადობის მიხედვით.

ბიჯი 3: თუ სტეკის თავზე მოთავსებული ჩანაწერი $[\underline{i}, L_F(\underline{x})]$ წარმოადგენს გზას ხის ბოლომდე, შევჩერდეთ და ავირჩიოთ $\hat{\underline{i}}_{[0,L+T)} = \underline{i}$. საწინააღმდეგო შემთხვევაში გადავიდეთ პირველ ბიჯზე.

წინა პარაგრაფში მოყვანილი მსჯელობის შემდეგ მკითხველისათვის ძნელი არ არის იმის დაჯერება, რომ აღწერილი სტეკ-ალგორითმი ახდენს თითქმის მაქსიმალური დამაჯერებლობით დეკოდირებას. მართლაც, შეიძლება ნაჩვენები იქნას, რომ შემთხვევითი ხისმაგვარი კოდების ანსამბლში

სტეკ-ალგორითმით შეცდომითი დეკოდირების ალბათობა აკმაყოფილებს (2.2) ფორმულით მოცემულ საზღვარს (უფრო ზუსტად, ვიტერბის ალგორითმთან შედარებით, სტეკ-ალგორითმისათვის ოდნავ მეტია „უმნიშვნელო“ მუდმივი სიდიდე - c_t [26]).

უნდა აღინიშნოს, რომ „ზოგად“ სტეკ-ალგორითმს არა აქვს განსაკუთრებული პრაქტიკული ღირებულება, ვინაიდან მე-2 ბიჯზე სტეკის ზომების მნიშვნელოვანი ზრდის გამო ძალზე იზრდება დეკოდირებისათვის განკუთვნილი დროც. ფ. ჯელინეკის მიერ აღწერილ ალგორითმის ვერსიაში, შეცდომის ალბათობის უმნიშვნელოდ ზრდის საფასურად, ხდება გამოთვლების დაჩქარება. კერძოდ, გამოყენებულია შემდეგი ხერხი: იგნორირებულია მეტრიკათა შორის სხვაობა მცირე წინასწარ განსაზღვრული კვანტირების Δ პარამეტრის შიგნით. უფრო ზუსტად, ფ. ჯელინეკმა განიხილა „კალათების“ სტეკი $...B_{-2}, B_{-1}, B_0, B_1, ...$ სადაც B_j შეიცავს სტეკში მოთავსებულ ყველა ისეთ ჩანაწერს, რომლისთვისაც სრულდება პირობა:

$$j\Delta \le L_F(\underline{x}) < (j+1)\Delta. \qquad (23.1)$$

ყველა ჩანაწერის „შენახვა“ და „ამოღება“ კალათებიდან ხორციელდება პრინციპით - „ბოლო შევიდა, პირველი გამოვიდა“. ფ. ჯელინეკის კალათების სტეკ-ალგორითმი შეიძლება აღიწეროს შემდეგნაირად:

ბიჯი 0: გავასუფთაოთ სტეკში არსებული ყველა კალათა და შემდეგ B_0 კალათაში შევიტანოთ ჩანაწერი $[\wedge, 0]$.

ბიჯი 1: სტეკის ყველაზე უფრო მაღლა მდებარე არაცარიელი კალათიდან ამოვიღოთ უკანასკნელად შეტანილი

ჩანაწერი $[\underline{i}, L_F(\underline{x})]$, გავაგრძელოთ ამ ჩანაწერის შესაბამისი გზა და მოვახდინოთ შემდეგი სიდიდეების ფორმირება: $[\underline{i} * 0, L_F(\underline{x}) + L_F(\underline{x}_0)]$ და $[\underline{i} * 1, L_F(\underline{x}) + L_F(\underline{x}_0)]$.

ბიჯი 2: შევინახოთ ორი ახლად ფორმირებული ჩანაწერი სათანადო კალათებში (23.1) ფორმულის შესაბამისად.

ბიჯი 3: თუ სტეკის ყველაზე უფრო მაღლა მდებარე არაცარიელ კალათში უკანასკნელად შეტანილი ჩანაწერი $[\underline{i}, L_F(\underline{x})]$ წარმოადგენს გზას ხის ბოლომდე, შევჩერდეთ და ავირჩიოთ $\hat{\underline{i}}_{[0,L+T)} = \underline{i}$. წინააღმდეგ შემთხვევაში გადავიდეთ პირველ ბიჯზე.

„ზოგადი“ სტეკ-ალგორითმისგან განსხვავებით დრო, რომელიც საჭიროა კალათების სტეკ-ალგორითმის მე-2 ბიჯის შესასრულებლად, არ არის დამოკიდებული სტეკში უკვე შეტანილი ჩანაწერების საერთო რიცხვზე, რაც საგრძნობლად აჩქარებს მის ფუნქციონირებას.

სტეკ-ალგორითმის პრაქტიკული რეალიზაციის სირთულე მნიშვნელოვნადაა დამოკიდებული დეკოდირების საბოლოო გადაწყვეტილების მიღებამდე ჩატარებულ „გამოთვლათა“ რაოდენობაზე. ჩატარებულ გამოთვლათა საზომ ერთეულად ავიღოთ ერთი კვანძის გაგრძელებისათვის საჭირო გამოთვლათა რიცხვი ანუ გამოთვლათა რაოდენობა, რომელიც საჭიროა ალგორითმის პირველი, მეორე და მესამე ბიჯების შესასრულებლად. აღსანიშნავია, რომ ვინაიდან თითოეული მომდევნო გამოთვლის ჩატარებისას ხდება სტეკიდან ერთი ჩანაწერის ამოღება და მასში ორი ახალი ჩანაწერის შეტანა, ამიტომ, დროის ნებისმიერ მომენტისათვის

შესრულებულ გამოთვლათა საერთო რაოდენობა ტოლია ამ მომენტში სტეკში არსებული ჩანაწერების საერთო რიცხვის. სტეკ-ალგორითმის მუშაობის უნარს ახასიათებენ C_0 სიდიდით, რომელსაც უწოდებენ გამოთვლით ძალისხმევას და იგი ტოლია მოცემული კოდური მიმდევრობის სადეკოდირებლად საჭირო გამოთვლათა რიცხვის ფარდობისა ამ მიმდევრობაში არსებულ საინფორმაციო ბიტების რაოდენობასთან. ამრიგად, გამოთვლითი ძალისხმევა ტოლია ერთ საინფორმაციო ბიტზე მოსულ გამოთვლათა რიცხვის. რა თქმა უნდა, C_0 შემთხვევითი სიდიდეა და მისი მნიშვნელობა დამოკიდებულია იმაზე, თუ რამდენად „ხმაურიანია“ მიღებული $\underline{y}_{[0,L+T)}$ მიმდევრობა. ამასთან, C_0 სიდიდე არ არის დამოკიდებული ხვევადი კოდის მეხსიერებაზე (რომლის ბაზაზეც აგებულია ხისმაგვარი დიაგრამა) და უმნიშვნელოდაა დამოკიდებული T სიდიდეზე. აქედან გამომდინარე, ჩვენ შეგვიძლია ავირჩიოთ M -ისა და T -ს ისეთი მნიშვნელობები, რომლებიც სტეკ-ალგორითმის (ან სხვა სახის მიმდევრობითი დეკოდირების ალგორითმის) გამოყენებისას შეცდომითი დეკოდირების ალბათობას გახდის ისე მცირეს, რომ შესაძლებელია მისი უგულებელყოფა. ამიტომ ერთადერთი ფაქტორი, რომელიც ზღუდავს ასეთი ალგორითმების მახასიათებლებს, ეს არის დეკოდირების დრო. თუ დეკოდირებისათვის განკუთვნილი დრო ნებას გვრთავს შევასრულოთ ყველაზე მეტი $n_{\max}$ გამოთვლა, მაშინ დეკოდირების შეწყვეტის ანუ წაშლის ალბათობა

$$P_{del} = P(C_0 > \frac{n_{\max}}{L}) \tag{23.2}$$

ხდება მნიშვნელოვანი პრაქტიკულად შემზღუდველი ფაქტორი. როგორც გამოკვლევები გვიჩვენებენ C_0 ძალზე არასასიამოვნო შემთხვევითი ცვლადია, რომელსაც უწოდებენ პარეტოს ცვლადს [5], [27]. მისთვის $P(C_0 > n)$ ალბათობა პროპორციულია n-ის მცირე უარყოფითი ხარისხისა, ამიტომ, მიმდევრობითი დეკოდირებისათვის არ ხერხდება P_{del} სიდიდის ძლიერ შემცირება და პრაქტიკაში გამოყენებული ასეთი სისტემისათვის ჩვეულებრივ $P_{del} = 10^{-3}$.

ამრიგად, მიმდევრობითი დეკოდირების გამოყენება რეკომენდებულია ისეთი მომხმარებლისათვის, რომელთაც (ა) აქვთ უკუკავშირის არხი და წაშლის შემთხვევაში ამ არხით აგზავნიან შეკითხვის სიგნალს, რომლის საშუალებითაც თავიდან იღებენ დაკარგულ ინფორმაციას, ან (ბ) არა აქვთ დიდი პრეტენზიები მონაცემთა მცირე რაოდენობის წაშლის შემთხვევაში, მაგრამ, სამაგიეროდ, ითხოვენ ძალზე მცირე შეცდომითი დეკოდირების ალბათობას იმ მონაცემებში, რომლებიც გადარჩენილია წაშლისაგან.

ახლა შევისწავლოთ ხისმაგვარ დიაგრამაზე იმ გზის ბუნება, რომელიც აირჩევა სტეკ-ალგორითმით (იგულისხმება, რომ მოცემულია საკმარისი დრო გამოთვლების დასრულებისათვის). აღვნიშნოთ V_j სიდიდით $(0 \le j \le L+T)$ მოცემული $\underline{i}$ გზის მეტრიკა ხის j სიღრმეზე, ანუ

$$V_j = L_F(\underline{x}_{[0,j)}), \tag{23.3}$$

სადაც $\underline{x}_{[0,j)}$ წარმოადგენს $\underline{i}_{[0,j)}$ საინფორმაციო მიმდევრობის შესაბამის კოდირებულ გზას, ანალოგიურად, აღვნიშნოთ V_j'

სიდიდით j სიღრმეზე რომელიმე სხვა $\underline{i}'$ გზის მეტრიკა. სამართლიანია შემდეგი ლემა:

ლემა 23.1 (არაარჩევითობის პრინციპი). თუ ხისმაგვარ დიაგრამაზე მოთავსებული $\underline{i}$ და $\underline{i}'$ გზები განშტოვდება j სიღრმეზე და

$$\min(V_{j+1}, V_{j+2}, ..., V_{L+T}) > \min(V'_{j+1}, V'_{j+2}, ..., V'_{L+T}), \qquad (23.4)$$

მაშინ სტეკ-ალგორითმით დეკოდირების ბოლოს $\underline{i}'$ არ შეიძლება წარმოადგენდეს სტეკის თავზე მოთავსებულ გზას.

დამტკიცება. თავდაპირველად შევნიშნოთ, რომ ვინაიდან $\underline{i}$ და $\underline{i}'$ განშტოვდებიან j სიღრმეზე, ამიტომ გვაქვს $\underline{i} = \underline{i}_{[0,j)} * \underline{i}_{[j,L+T)}$ და $\underline{i}' = \underline{i}'_{[0,j)} * \underline{i}'_{[j,L+T)}$. აქედან გამომდინარე, არც $\underline{i}$ და არც $\underline{i}'$ არ შეიძლება იყოს საბოლოოდ არჩეული გზა, თუ დროის რომელიმე მომენტისათვის ჩანაწერი $[\underline{i}_{[0,j)}, V_j]$ მიაღწევს სტეკის ზედა საზღვარს და, ამიტომ, დროის მომდევნო მომენტში არ იქნება გაგრძელებული მისი შესაბამისი გზა, რაც, თავის მხრივ, გამოიწვევს სტეკში ორი ახალი ჩანაწერის $[\underline{i}_{[0,j+1)}, V_{j+1}]$ და $[\underline{i}'_{[0,j+1)}, V'_{j+1}]$ შეტანას. და თუ ეს მოხდა, ამ მომენტის შემდეგ სტეკში ყოველთვის იქნება მოთავსებული $\underline{i}_{[0,j+1]}$ გზა ან მისი გაგრძელებები. ასე რომ, სტეკში ყოველთვის იქნება ჩანაწერი, რომლის V მეტრიკა აკმაყოფილებს პირობას:

$$V \geq \min(V_{j+1}, V_{j+2}, ..., V_{L+T}). \qquad (23.5)$$

ახლა დავუშვათ, რომ V'_{j+k} წარმოადგენს მინიმალურ სიდიდეს $V'_{j+1}, V'_{j+2}, ..., V'_{L+T}$ მეტრიკებს შორის. თუ $\underline{i}'$ აღწევს

სტეკის ზედა საზღვარს, მაშინ $\underline{i}'_{[0,j+k)}$-ს ადრე უნდა მიეღწია სტეკის ზედაპირისათვის, ვინაიდან $\underline{i}'$ წარმოადგენს ამ გზის გაგრძელებას. მაგრამ (23.4) და (23.5) ფორმულების თანახმად $V'_{j+k} < V$, ასე რომ, $\underline{i}'_{[0,j+k)}$-ს არ შეეძლო მიეღწია სტეკის ზედა საზღვრისათვის, ვინაიდან სტეკში ყოველთვის არსებობს ჩანაწერი უფრო მაღალი მეტრიკით. ამრიგად, ვასკვნით, რომ $\underline{i}'$ არ შეიძლება იყოს არჩეული გზა. □

სტეკ-ალგორითმით დეკოდირების არაარჩევითობის პრინციპი, ისევე როგორც მაქსიმალური დამაჯერებლობით დეკოდირების არაოპტიმალურობის პრინციპი, შეიძლება გამოყენებულ იქნას ხისმაგვარ დიაგრამაზე „საუკეთესო“ გზის ასარჩევად. მოქმედება შეიძლება დავიწყოთ $L-1$ სიღრმეზე და ამ სიღრმეზე მოთავსებულ ყველა კვანძში უგულვებელვყოთ ორი გამოსული გზიდან ის, რომელსაც აქვს $\min(V_L, V_{L+1}, ..., V_{L+T})$ სიდიდის უმცირესი მნიშვნელობა. შემდეგ გადავდივართ $L-2$ სიღრმეზე და ამ სიღრმეზე ყველა დატოვებულ კვანძში უგულვებელვყოფთ ორი გამოსული გზიდან ერთ მათგანს $\min(V_L, V_{L+1}, ..., V_{L+T})$ უმცირესი მნიშვნელობით და ა. შ. დროის გარკვეული პერიოდის გავლის შემდეგ (ვთქვათ, $L = 50$) ჩვენ მივაღწევთ სიღრმეს 0-ს და გზების უგულებელყოფის აღწერილი პროცედურა დაგვიტოვებს ერთადერთ გზას, სახელდობრ, გზას, რომელიც არჩეული იქნებოდა სტეკ-ალგორითმით დეკოდირებისას. მეორე მხრივ, ჩვენ შეგვიძლია უშუალოდ გამოვიყენოთ სტეკ-ალგორითმი ხისმაგვარ დიაგრამაზე საკუთარი გზის ასარჩევად, ვიწყებთ რა დეკოდირებას ხის ფესვური კვანძიდან. ამრიგად,

არაოპტიმალურობის პრინციპისაგან განსხვავებით, (23.4) ფორმულას აქვს მხოლოდ აზრობრივი მნიშვნელობა იმ გზის ბუნების აღსაწერად, რომელიც საბოლოოდ არჩეული იქნება მიმდევრობითი დეკოდირების რომელიმე ალგორითმის გამოყენებით და იგი უშუალოდ არ გვთავაზობს კონკრეტულ ალგორითმს.

არაარჩევითობის პრინციპიდან გამომდინარე, ცხადია, რომ სტეკ-ალგორითმით შეცდომითი დეკოდირების P_e ალბათობა იქნება მცირე მაშინ და მხოლოდ მაშინ, როდესაც მეტრიკას აქვს შემდეგი თვისებები: იგი უნდა იზრდებოდეს ხისმაგვარ დიაგრამაზე სინამდვილეში გადაცემული სწორი გზის გასწვრივ და მცირდებოდეს ყველა იმ გზის გასწვრივ, რომლებიც განშტოვდება სწორი გზისაგან. სწორედ ეს იყო ის მიზეზი, რომელმაც რ. ფანო მიიყვანა (22.5) ფორმულით მოცემულ გამოსახულებამდე. რ. ფანო თვლიდა, რომ პირველი წევრი მეტრიკის გამოსახულებაში წარმოადგენს ერთობლივ ინფორმაციას მიღებულ y_n სიმბოლოსა და ჰიპოთეზის ქვეშ მყოფ x_{sn} სიმბოლოს შორის. თუ მართლაც, გადაცემული იყო x_{sn} მაშინ ამ ერთობლივი ინფორმაციის საშუალო მნიშვნელობა უნდა ყოფილიყო არხის გამტარუნარიანობა C და სწორი გზის თითოეული სიმბოლოს დამუშავებისას მეტრიკა საშუალოდ უნდა გაზრდილიყო დადებითი $C - R$ წანაზარდით. არასწორ გზაზე x_{sn} სიმბოლო სტატისტიკურად დამოუკიდებელი უნდა ყოფილიყო y_n-ზე და ამიტომ მათ შორის ერთობლივი ინფორმაციის საშუალო მნიშვნელობა ტოლი უნდა ყოფილიყო ნულის. აქედან გამომდინარე, არასწორი

გზის მეტრიკა საშუალოდ უნდა შემცირებულიყო უარყოფითი $-R$ წანაზარდით ამ გზის თითოეული სიმბოლოს დამუშავებისას.

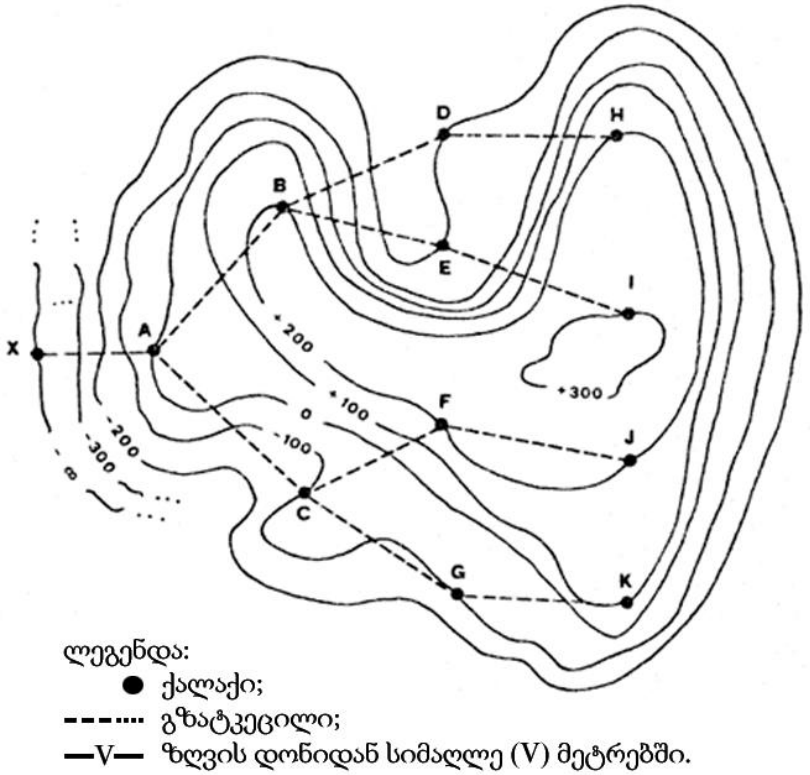
24. მიმდევრობითი დეკოდირება - ფანოს ალგორითმი

სტეკ-ალგორითმი საჭიროებს ინფორმაციის დიდი რაოდენობით დამახსოვრებას. იგი იყენებს ამ ინფორმაციას ხისმაგვარი დიაგრამის „ეფექტურად“ გამოსაკვლევად, რაც იმას ნიშნავს, რომ თითოეული გამოსაკვლევი კვანძი მუშავდება ანუ „გრძელდება“ მხოლოდ ერთხელ. რ. ფანოს მიერ შესწავლილი ალგორითმი [23] წარმოადგენს ხისმაგვარ დიაგრამაზე გზის მოძებნის ორიგინალურ მეთოდს, რომელიც ფუნქციონირებს წინა პარაგრაფში ჩამოყალიბებული არაარჩევითობის პრინციპის მიხედვით და, ამასთან, თითქმის არ იყენებს მეხსიერებას. ფანოს ალგორითმით დეკოდირებისას დაშვებულია ერთსა და იმავე კვანძის მრავალჯერ დამუშავება და, მაშასადამე, მისი მრავალჯერ გაგრძელებაც. ამასთან, მთელი ცოდნა ადრე ჩატარებული გამოკვლევების შესახებ დაყვანილია მხოლოდ ერთი რიცხვის დამახსოვრებაზე.

ფანოს ალგორითმი იოლად აღიქმება შემდეგი ანალოგიის ბაზაზე. განვიხილოთ 21-ე ნახაზზე მოცემული რუკა. ეს რუკა წარმოგვიდგენს ქალაქიდან გამოსული გზატკეცილების ხეს და A ქალაქი მდებარეობს ხის „ფესვურ კვანძში“ (საიდუმლო X ქალაქი, რომელიც მდებარეობს ზღვის დონიდან ძალზე დაბლა, ხელოვნურად გამოგონილი ქალაქია და მისი

საჭიროება ცნობილი გახდება შემდგომი მსჯელობისას; ამჯერად კი ვიგულისხმოთ, რომ იგი მოთავსებულია ფესვური A კვანძიდან სიღრმეზე (-1)). ვთქვათ, მოგზაურს სურს გადაადგილდეს A ქალაქიდან ერთ-ერთი იმ საბოლოო H, I, J, K ქალაქისკენ, რომელიც მოძებნილი იქნებოდა სტეკ-ალგორითმით. ამრიგად, მას სურს იმოგზაუროს ისეთი V_0, V_1, V_2, V_3 გზით, რომელიც გაიმარჯვებს (23.4) ფორმულით მოცემულ უტოლობაში ნებისმიერ სხვა, მისგან განშტოებულ გზაზე. თუ ვისარგებლებთ წინა პარაგრაფში განხილული სტეკ-ალგორითმით, სტეკის შემცველობა თითოეული გამოთვლის შემდეგ იქნება: (იხ. 21-ე ნახაზი).

ვინაიდან J საბოლოო ქალაქია, სადაც მთავრდება მოგზაურობა და ამასთან, იგი მოთავსებულია სტეკის ზედა საზღვართან, ამიტომ J იქნება ის ქალაქი, რომელსაც მიაღწევს მოგზაური სტეკ-ალგორითმის მიხედვით გადაადგილებისას (აქ უნდა აღინიშნოს, რომ J არ არის ზღვის დონიდან ყველაზე უფრო მაღლა მდებარე ქალაქი და თუ მოგზაური იმოძრავებდა მაქსიმალური დამაჯერებლობის დეკოდერის შესაბამისად, ის საბოლოოდ მიაღწევდა I ქალაქს). ამრიგად, ჩვენ ვხედავთ, რომ სტეკ-ალგორითმის გამოყენებისას მოგზაურმა აუცილებლად უნდა „დაიმახსოვროს“ ზღვის დონიდან ყველა იმ ქალაქის სიმაღლე, რომელიც მდებარეობს მისი გადაადგილების მარშრუტზე.



გამოთვლათა ნომერი	1	2	3	4	5
სტეკის შემცველობა	[A, 0]	[B, 200] [C, -100]	[C, -100] [D, -200] [E, -200]	[F, 200] [G, -100] [D, -200] [E, -200]	[J, 200] [G, -100] [D, -200] [E, -200]

ნახ. 21. კონტურული რუკა, რომელიც გამოყენებულია ფანოს ალგორითმით ხისმაგვარი დიაგრამის გამოკვლევის საილუსტრაციოდ და სტეკის შემცველობის ცხრილი

მეორე მხრივ, (23.4) უტოლობიდან გამომდინარე, მოგზაური თითოეული განშტოების გავლისას ირჩევს გზატკეცილს საუკეთესო ქალაქამდე მანამ, სანამ ეს ქალაქი რჩება რომელიმე კრიტერიუმით განსაზღვრულ ,,მისაღებ“ გზაზე დიაგრამაში.

დავუშვათ, რომ სხვა მოგზაური იყენებს „დასაშვებ T ზღუდეს“ იმის გადასაწყვეტად, თუ დროის რა მომენტში აღარ კმაყოფილდება (23.4) ფორმულით მოცემული უტოლობა იმ გზისათვის, რომლის გასწვრივაც ის მოძრაობს. ასე რომ, იგი უნდა დაბრუნდეს უკან და გასინჯოს სხვა გზა. ეს მოგზაურიც თავდაპირველად გადაადგილებას იწყებს A ქალაქიდან და აყენებს $T = V_A = 0$. ის ხედავს, რომ ამაღლება წინ საუკეთესო ქალაქისაკენ ტოლია $V_A = 200$. ეს აკმაყოფილებს მისი ზღუდის ტესტს და, მაშასადამე, იგი გადაადგილდება B ქალაქისაკენ. ახლა მოგზაური ზრდის თავის ზღუდეს $T = V_B = 200$ მნიშვნელობამდე, ვინაიდან იმედოვნებს, რომ აღარასოდეს ჩამოვა უფრო დაბალ მნიშვნელობამდე. შემდეგ იგი იყურება წინ და ხედავს, რომ არც ერთი წინ მდებარე ქალაქი არ აკმაყოფილებს მის ზღუდეს, მაშინ მოგზაური იყურება უკან და ხედავს, რომ ქალაქი, საიდანაც ის ჩამოვიდა, ასევე არ აკმაყოფილებს მის ზღუდეს. ამრიგად, მას არ დარჩენია სხვა გამოსავალი გარდა იმისა, რომ შეამციროს თავისი ზღუდე $T = 100$ მნიშვნელობამდე. ამის შემდეგ ის იყურება წინ D და E ქალაქებისაკენ და ხედავს, რომ მისი იქ გადაადგილებისას ზღუდე ისევე დაირღვევა. ამიტომ, იგი იყურება უკან A ქალაქისაკენ, მაგრამ მისი ზღუდის ტესტი ამჯერადაც არ კმაყოფილდება. მაშასადამე, მოგზაურმა ისევ უნდა

შეამციროს ზღუდის მნიშვნელობა, ამჯერად $T = 0$ სიდიდემდე. შემდეგ ის ისევ იყურება წინ D და E ქალაქებისაკენ, მაგრამ მას ისევ არ შეუძლია არც ერთ მათგანში შესვლა. მაშინ მოგზაური იყურება უკან A ქალაქისაკენ და ხედავს, რომ მას შეუძლია უკან მოძრაობა და იგი ბრუნდება A ქალაქში.

ამრიგად, მოგზაურობამ საუკეთესო B ქალაქისაკენ მას არ მოუტანა არავითარი შედეგი და ამიტომ მოგზაური იყურება სხვა მიმართულებით, მეორე გზატკეცილზე მდებარე C ქალაქისაკენ, მაგრამ ისევ ხედავს, რომ არც იქით შეუძლია გადაადგილება. მაშინ ის იყურება უკან „საიდუმლო“ X ქალაქისაკენ და ხედავს, რომ ამ ქალაქში შესვლაც მიუწვდომელი ოცნებაა მისთვის. ამრიგად, მოგზაური მოექცა ხაფანგში და რწმუნდება, რომ არ არსებობს A ქალაქიდან გამომავალი არცერთი გზატკეცილი, რომელიც რჩება 0 სიმაღლეზე ან იმყოფება უფრო მაღლა ამ გზატკეცილის ყველა წერტილში. აქედან გამომდინარე, მოგზაური იძულებულია შეამციროს ზღუდე $T = -100$ მნიშვნელობამდე, შემდეგ იგი იყურება წინ და გადაადგილდება საუკეთესო B ქალაქში. მაგრამ რა უნდა უყოს ამჯერად მან თავის ზღუდეს? (მოგზაურმა უკვე დაივიწყა, მაგრამ ჩვენ კი ვიცით, რომ თუ ის ისევ ასწევს თავის ზღუდეს $V_B = 200$ მნიშვნელობამდე, იგი მუდმივად იმოძრავებს ჩაკეტილ მარყუჟში). ერთადერთი, რისი ცოდნაც შეუძლია მიგზაურს - ეს არის ის, რომ მისი ზღუდე არ იყო კომპაქტური A ქალაქში, ანუ მიუხედავად იმისა, რომ A ქალაქის სიმაღლე ტოლია 0-ის, მისი იქიდან გამოსვლისას ზღუდის მნიშვნელობა შეესაბამებოდა $T = -100$-ს. ეს ფაქტორი მასზე მოქმედებს იმის მანიშნებლად, რომ ხელი არ ახლოს ზღუდეს. ამრიგად, ის ტოვებს ზღუდის მნიშვნელობას

$T = -100$. შემდეგ მოგზაური იყურება წინ და ხედავს, რომ არ შეუძლია D ან E ქალაქისაკენ გადაადგილება. იგი იყურება უკან და ხედავს, რომ შეუძლია A ქალაქში დაბრუნება, რასაც ის აკეთებს. როდესაც მიხვდა, რომ იგი დაბრუნდა უკეთესი გზატკეცილიდან, მოგზაური ცდილობს სცადოს თავისი ბედი C ქალაქის მიმართულებით. ის ამჩნევს, რომ შეუძლია C ქალაქისაკენ გადაადგილება და ვინაიდან მისი ზღუდე $T = -100$ უკვე კომპაქტურია, იგი არ ცვლის მას. C ქალაქიდან მოგზაური იყურება წინ და ხედავს, რომ უკეთეს გზატკეცილს მიჰყავს F ქალაქში. იგი გადაადგილდება F ქალაქში და რადგანაც მისი ზღუდე C ქალაქში იყო კომპაქტური, ეს უფლებას აძლევს გაზარდოს ზღუდის მნიშვნელობა და გაუტოლოს ის $T = V_F = 200$ -ს. ამის შემდეგ მოგზაური იყურება წინ და ხედავს, რომ შეუძლია გადაადგილება J ქალაქისაკენ, რასაც ის აკეთებს. ვინაიდან J საბოლოო ქალაქია, იგი ამთავრებს თავის მოგზაურობას. ამრიგად, მოგზაური ჩავიდა იგივე J ქალაქში, რომელსაც საბოლოოდ მიაღწია სტეკ-ალგორითმით ორიენტირებულმა მოგზაურმა.

ამ მეორე მოგზაურმა ხისმაგვარი დიაგრამის გამოკვლევისათვის გამოიყენა ფანოს ალგორითმი [23]. ალგორითმის მუშაობის პრინციპს ყველაზე კარგად ხსნის 22-ე ნახაზზე მოცემული ბლოკ-სქემა.

დავუშვათ, რომ ხისმაგვარ დიაგრამაზე ფესვური კვანძის უკან მოთავსებულია „გამოგონილი“ კვანძი $(-\infty)$ მეტრიკით, ასე რომ ფესვური კვანძიდან უკან გადაადგილება ყოველთვის გამოიწვევს ზღუდის შემცირებას. ბლოკ-სქემაში V_3 აღნიშნავს მეტრიკის მნიშვნელობას ფესვური კვანძიდან პირ-

დაპირი მიმართულებით მდებარე გზაზე, ე. ი. ამ გზის რომელიმე კვანძიდან შემდგომ სიღრმეზე მდებარე შესასწავლ კვანძამდე, ხოლო $V_{უ}$ აღნიშნავს მეტრიკის მნიშვნელობას უკუმიმართულებით ფესვური კვანძისაკენ მიმავალ გზაზე, ე. ი. ამ გზის რომელიმე კვანძიდან წინა სიღრმეზე მდებარე კვანძამდე. „გახდეს ზღუდე კომპაქტური“ ნიშნავს T გაიზარდოს $z\Delta$ მნიშვნელობით, სადაც z უდიდესი მთელი რიცხვია, რომლისთვისაც მეტრიკის V მნიშვნელობა ფესვური კვანძიდან მოცემულ მომენტში დაკავებულ კვანძამდე ჯერ კიდევ არ არღვევს ზღუდის მნიშვნელობას.

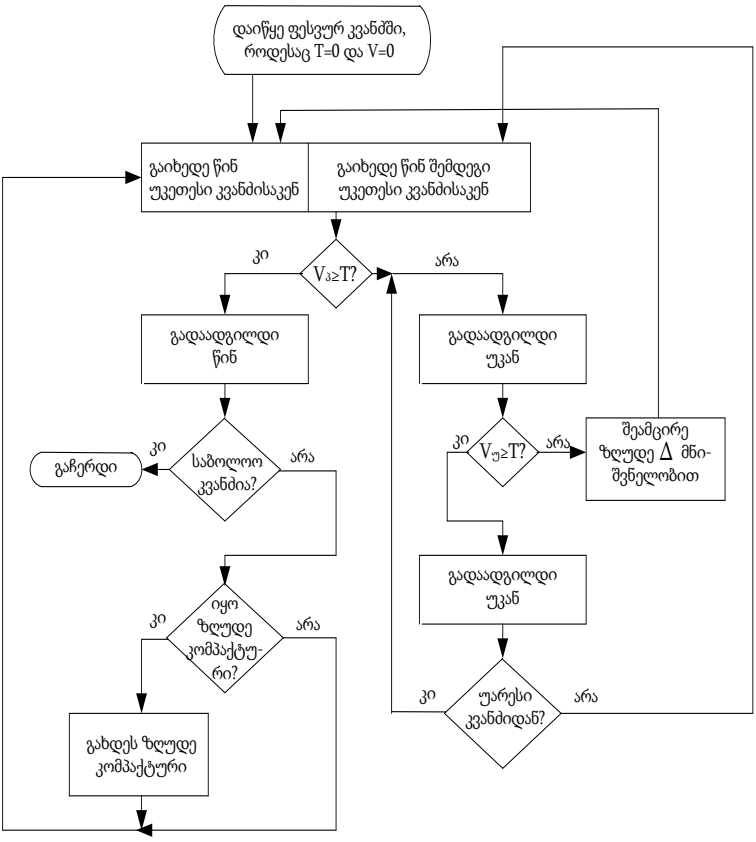
ამრიგად, ფანოს ალგორითმით დეკოდირების პროცესში წინ მოძრაობისას არც ერთი კვანძი არ გამოიკვლევა ორჯერ ზღუდის ერთსა და იმავე მნიშვნელობისათვის. ნებისმიერი კვანძის ყოველი შემდგომი გამოკვლევისას ზღუდის მნიშვნელობა უნდა იყოს ნაკლები, ვიდრე ამ კვანძის წინა გამოკვლევისას.

ძნელი არ არის ვუჩვენოთ, რომ ფანოს ალგორითმი ყოველთვის მოძებნის იგივე გზას ხისმაგვარ დიაგრამაში, რასაც სტეკ-ალგორითმი, თუ სრულდება შემდეგი პირობები: (ა) ყველა კვანძის მეტრიკის მნიშვნელობა წარმოადგენს Δ -ს ჯერადს და (ბ) გამარჯვებულ გზას არა აქვს „კუდი“, ე. ი. სტეკ-ალგორითმის არაარჩევითობის პრინციპის მიხედვით შეიძლება არ იყოს ლიკვიდირებული მხოლოდ ზუსტად ერთი გზა [28]. განხილული მოგზაურის მაგალითი ნათელყოფს, თუ რატომ სრულდება არაარჩევითობის პრინციპი ფანოს ალგორითმისთვისაც. ზღუდე იწევს 0-ზე დაბლა მხოლოდ ფესვურ კვანძში და მხოლოდ ეს იძლევა იმის გარანტიას, რომ

საბოლოოდ შეუძლია გაიმარჯვოს გზამ, რომელსაც აქვს უდიდესი $\min(V_1, V_2, ..., V_{L+T})$ მნიშვნელობა. ნებისმიერ სხვა კვანძში პირველად სტუმრობისას ზღუდე უნდა იყოს კომპაქტური (ისევე, როგორც თავდაპირველად ფესვურ კვანძში), განსხვავებით ამავე კვანძში შემდგომი სტუმრობისა, თანაც იგი უკვე კომპაქტურია მოგზაურის იქ პირველად მოხვედრისას, ან ხდება კომპაქტური „იყო ზღუდე კომპაქტური?“ ტესტით შემოწმებისას. ამრიგად, ყველა დანარჩენ კვანძში პირველად სტუმრობისასაც ადგილი აქვს ისეთივე სიტუაციას, როგორსაც ფესვურ კვანძში და არაარჩევითობის ზემოთ მოტანილი არგუმენტი ფესვური კვანძისათვის სამართლიანია ნებისმიერი სხვა კვანძისათვისაც. ამრიგად, ვრწმუნდებით, რომ ფანოს ალგორითმით არჩეული გზა არ შეიძლება იყოს ისეთი, რომელიც ლიკვიდირებული იქნებოდა არაარჩევითობის პრინციპით. ვინაიდან, დაშვების თანახმად, არსებობს მხოლოდ ერთი ასეთი გზა, მარტივად შეიძლება შემოწმდეს, რომ ფანოს ალგორითმი ყოველთვის მოძებნის იმავე გზას, რასაც კალათების სტეკ-ალგორითმი და კვანტირების Δ პარამეტრი ამ უკანასკნელისათვის თამაშობს იმავე როლს, რომელსაც ასრულებდა ზღუდის მნიშვნელობასთან დაკავშირებული Δ სიდიდე ფანოს ალგორითმისათვის.

ჯ. გეისტის მიერ შემუშავებულ იქნა სტეკ-ალგორითმის და ფანოს ალგორითმის მშვენიერი პროცედურები მათი პროგრამული რეალიზაციისათვის [29]. ერთ-ერთი პროგრამული „ხერხი“, რაც მნიშვნელოვნად ზრდის გამოთვლების სიჩქარეს, მდგომარეობს იმაში, რომ V და T სიდიდეების ცალ-ცალკე დამახსოვრების ნაცვლად გამოყენებულია მათი

სხვაობა $Q = V - T$ და ფანოს ალგორითმთან დაკავშირებული ყველა ტესტი ტარდება Q სიდიდის მიხედვით.



ნახ. 22. ფანოს მიმდევრობითი დეკოდირების ალგორითმის ბლოკ-სქემა

უნდა აღინიშნოს, რომ სტეკ-ალგორითმი საჭიროებს საკმაოდ დიდ მეხსიერებას, მაგრამ, ამასთან, მის მიერ დამახ-

სოვრებული მონაცემების ძალზე მცირე ლოგიკურ დამუშავებას. რაც შეეხება ფანოს ალგორითმს, თუ Δ სიდიდე სათანადოდაა შერჩეული, იგი წარმოადგენს მიმდევრობითი დეკოდირების ეფექტურ პროცედურას, განსაკუთრებით მაშინ, როდესაც მცირე მეხსიერებაა ხელმისაწვდომი. ორივე ალგორითმი კარგად უთანხმდება თანამედროვე ელექტრონულ გამომთვლელ მანქანებს და, როგორც გამოკვლევები გვიჩვენებენ, შედარებით დაბალი სიჩქარეებისათვის $(R \le 0.9R_0)$ ფანოს ალგორითმი ახდენს უფრო სწრაფ დეკოდირებას, ვიდრე სტეკ-ალგორითმი, ხოლო მაღალი სიჩქარეებისათვის მიზანშეწონილია სტეკ-ალგორითმის გამოყენება. დასასრულს უნდა აღინიშნოს ისიც, რომ საზოგადოდ მიმდევრობითი დეკოდირების ალგორითმები ეფექტურად ფუნქციონირებს მხოლოდ შედარებით „უხმაურო“ კავშირის არხებში, ხოლო არხებში ცუდი მახასიათებლებით, როგორც წესი, იყენებენ ვიტერბის ალგორითმს.

25. ხვევადი კოდების ზოგიერთი კლასი

განვიხილოთ ზოგიერთი ტიპის ხვევადი კოდი მუდმივი ხვევადი კოდერებით.

ორობითი მუდმივი ხვევადი კოდერი $R = 1/2$ სიჩქარით და M მეხსიერებით წარმოვადგინოთ გადამცემი ფუნქციის მატრიცის შემადგენელი წარმომქმნელი მრავალწევრების საშუალებით:

$$G^{(i)}(D) = g_0^{(i)} + g_1^{(i)}D + g_2^{(i)}D^2 + ... + g_M^{(i)}D^M ,$$

სადაც $g_j^{(i)}$, $1 \le i \le N$, ორობითი სიმბოლოა. ქვემოთ $G^{(i)} = \left\{ g_0^{(i)}, g_1^{(i)}, ..., g_M^{(i)} \right\}$ სიდიდეები ჩაწერილია რვაობითი ფორმით. მაგალითად, $G^{(1)} = 554$ და $G^{(2)} = 744$ ჩანაწერი შეესაბამება ორობით მუდმივ ხვევად კოდერს წარმომქმნელი მრავალწევრებით:

$$G^{(1)}(D) = 1 + D^2 + D^3 + D^5 + D^6,$$

$$G^{(2)}(D) = 1 + D + D^2 + D^3 + D^6.$$

მე-3, მე-4 და მე-5 ცხრილებში მოყვანილია არაკატასტროფული მუდმივი ხვევადი კოდერები შესაბამისად სიჩქარეებით $1/2, 1/3, 1/4$ და მეხსიერებით $2 \le M \le 13$, რომელთაც აქვთ თავისუფალი მანძილის (d_∞) მაქსიმალური (დღეისათვის ცნობილი) მნიშვნელობა. აქვე მოცემულია ამ კოდების თავისუფალი მანძილის ზედა საზღვარი - d_u. ეს კოდები აგებული იყო კ. ლარსენის მიერ [15]. მე-3, მე-4 და მე-5 ცხრილებში მოყვანილ კოდერებს აქვთ დაბალი სიჩქარეები $R \le 1/2$. უფრო მაღალი სიჩქარის კარგი ხვევადი კოდერები შეიძლება მიღებულ იქნან როგორც უშუალო გადარჩევით [30], ასევე დაბალი სიჩქარის კოდერებში მიმდევრობების სიმბოლოების გარკვეული წესით პერფორაციის (ამოგდების) საშუალებით. ეს მეთოდი ძალზე ეფექტურია, ვინაიდან ის გვაძლევს ოპტიმალურთან ახლოს მდგარი კოდერების ფართო კლასს (ზოგიერთი მათგანი საუკეთესოცაა თავისი პარამეტრებით) და, ამასთან, მათი დეკოდირების სირთულე პრაქტიკულად ისეთივეა, როგორიც საწყისი კოდის დეკოდირების სირთულე.

ცხრილი 3

M	$G^{(1)}$	$G^{(2)}$	d_∞	d_u
2	5	7	5	5
3	64	74	6	6
4	46	72	7	8
5	53	75	8	8
6	554	744	10	10
7	516	762	10	11
8	561	753	12	12
9	4734	6624	12	13
10	4672	7542	14	14
11	4335	5723	15	16
12	42554	77304	16	16
13	43572	76246	16	17

ცხრილი 4

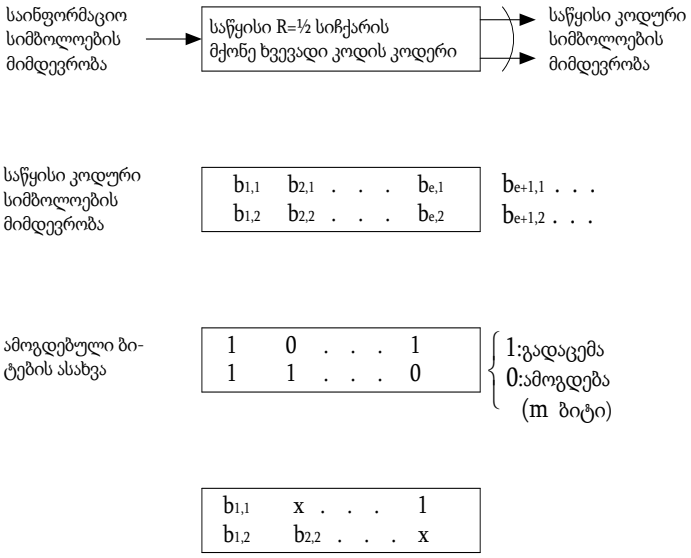
M	$G^{(1)}$	$G^{(2)}$	$G^{(3)}$	d_∞	d_u
2	5	7	7	8	8
3	54	64	74	10	10
4	54	66	76	12	12
5	47	53	75	13	13
6	554	624	764	15	15
7	452	662	756	16	16
8	557	663	711	18	18
9	4474	5724	7154	20	20
10	4726	5562	6372	22	22
11	4767	5723	6265	24	24
12	42554	43364	77304	24	24
13	43512	73542	76266	26	26

ცხრილი 5

M	$G^{(1)}$	$G^{(2)}$	$G^{(3)}$	$G^{(4)}$	d_{∞}	d_u
2	5	7	7	7	10	10
3	54	64	64	74	15	15
4	52	56	66	76	16	16
5	53	67	71	75	18	18
6	564	564	534	714	20	20
7	472	572	626	736	22	22
8	463	535	733	745	24	24
9	4474	5724	7154	7254	27	27
10	4656	4726	5562	6372	29	29
11	4767	5723	6265	7455	32	32
12	44624	52374	66754	73534	33	33
13	42226	46372	73256	73276	36	36

23-ე ნახაზზე მოცემულია $R \le 1/2$ სიჩქარის მქონე კოდებისაგან უფრო მაღალი სიჩქარის კოდების მიღების ზოგადი პროცედურა. ძირითადი საწყისი კოდური მიმდევრობის l ბლოკში (ე. ი. $2l$ კოდურ ბიტში) m ბიტი პერიოდულად ამოიკვეთება და არ გადაიცემა. ეს პროცედურა ხორციელდება წინასწარ განსაზღვრული სქემის მიხედვით, რომელიც გვიჩვენებს, თუ, სახელდობრ, რომელი ბიტები უნდა იქნას ამოგდებული. როდესაც m-ს ვირჩევთ $l-1$-ის ტოლს, მიიღება პერფორირებული კოდი $R = (N-1)/N$ სიჩქარით. მე-6 ცხრილში მოცემულია პერფორირებული კოდები, მიღებული პირველ ცხრილში წარმოდგენილი $R = 1/2$ სიჩქარის მქონე ხვევადი კოდებისაგან, ამასთან მითითებულია პერფორირებული კოდების მიღების მეთოდი და მოცემულია მათი თავისუფალი მანძილი [31]. როგორც ცხრილიდან ჩანს, მართალია ფიქსირებული მეხსიერებისათვის სიჩქარის გაზრდისას თა-

ვისუფალი მანძილის მნიშვნელობა ყოველთვის არ მცირდება, მაგრამ მხედველობაში მისაღებია ის ფაქტიც, რომ ამ დროს იზრდება d_{∞} ჰემინგის წონიანი გზათა რიცხვი, რაც აუარესებს კოდის ხელშეშლამდგრადობას.



ნახ. 23. პერფორირებული კოდების მიღების ზოგადი პროცედურა $R = 1/2$ სიჩქარის მქონე ხვევადი კოდებისაგან (×-ამოგდებული ბიტები)

მე-7 ცხრილში მოყვანილია ისეთი არაკატასტროფული ხვევადი კოდერები, რომელთაც აქვთ:

1. ოპტიმალური დისტანციური პროფილი;
2. ოპტიმალური თავისუფალი მანძილი პირველი პირობის დაცვით;

ცხრილი 6

M / R	2		3		4	
	ამოგდებული ბიტები	d_∞	ამოგდებული ბიტები	d_∞	ამოგდებული ბიტები	d_∞
1/2	1 (5) 1 (7)	5	1 (64) 1 (74)	6	1 (46) 1 (72)	7
2/3	10 11	3	11 10	4	11 10	4
3/4	101 110	3	110 101	4	101 110	3
4/5	1011 1100	2	1011 1100	3	1010 1101	3
5/6	10111 11000	2	10100 11011	3	10111 11000	3
6/7	101111 110000	2	100011 111100	2	101010 110101	3
7/8	1011111 1100000	2	1000010 1111101	2	1010011 1101100	3
8/9	10111111 11000000	2	10000011 11111100	2	11100010 10011101	3
9/10	101111111 110000000	2	101000000 110111111	2	100001111 111110000	2
10/11	1011111111 1100000000	2	1000000011 1111111100	2	1000000101 1111111010	2
11/12	10111111111 11000000000	2	10000000010 11111111101	2	10101101101 11010010010	2
12/13	101111111111 110000000000	2	100000000011 111111111100	2	101101111011 110010000100	2
13/14	1011111111111 1100000000000	2	1010000000000 1101111111111	2	1110110110111 1001001001000	2

ცხრილი 6 (გაგრძელება)

M / R	5		6		7	
	ამოგდებუ-ლი ბიტები	d_∞	ამოგდებუ-ლი ბიტები	d_∞	ამოგდებუ-ლი ბიტები	d_∞
1/2	1 (53) 1 (75)	8	1 (554) 1 (744)	10	1 (516) 1 (762)	10
2/3	10 11	6	11 10	6	11 10	7
3/4	100 111	4	110 101	5	110 101	6
4/5	1000 1111	4	1111 1000	4	1010 1101	5
5/6	10000 11111	4	11010 10101	4	11100 10011	4
6/7	110110 101001	3	111010 100101	3	101001 110110	4
7/8	1011101 1100010	3	1111010 1000101	3	1010100 1101011	4
8/9	11100010 10011101	3	11110100 10001011	3	10110110 11001001	3
9/10	100001111 111110000	3	111101110 100010001	3	101100110 110011001	4
10/11	1001110100 1110001011	3	1110110111 1001001000	3	1001000011 1110111100	3
11/12	10001110100 11110001011	3	11110111110 10001000001	3	10110000110 11001111001	3
12/13	110100110110 101011001001	3	111111110101 100000001010	3	100100001100 111011110011	3
13/14	1100011000100 1011100111011	3	1101000001111 1010111110000	3	1010100100000 1101011011111	3

3. d_∞ ჰემინგის წონიანი გზების მინიმალური რაოდენობა მეორე პირობის დაცვით.

ცხრილი 7

M	$G^{(1)}$	$G^{(2)}$	d_M	გზების რიცხვი წონით d_M	d_∞	გზების რიცხვი წონით d_∞
1	6	4	3	2	3	1
2	7	5	3	1	5	1
3	74	54	4	3	6	1
4	62	56	4	2	7	2
5	75	55	5	6	8	2
6	634	564	5	3	10	12
7	626	572	6	11	10	1
8	751	557	6	6	12	10
9	7664	5714	6	2	12	1
10	7512	5562	7	13	14	19
11	6643	5175	7	5	14	1
12	63374	47244	8	29	15	2
13	45332	77136	8	12	16	5
14	65231	43677	8	10	17	3
15	517604	664134	8	5	18	10
16	717066	522702	9	18	19	9
17	506477	673711	9	7	21	13
18	5653664	7746714	9	7	21	13
19	5122642	7315626	10	31	22	26
20	6567413	5322305	10	13	22	2
21	6752065	50371444	10	4	24	10
22	67132702	50516146	10	1	24	25
23	55346125	75744143	11	28	25	13

ასეთი კოდერებიდან ოპტიმალური თავისუფალი მანძილი აქვთ მხოლოდ კოდებს მეხსიერებით $M = 1,...,10$ და $M = 13$. მე-8 ცხრილში წარმოდგენილია ზოგიერთი სხვა არაკატასტროფული მუდმივი ხვევადი კოდერი, რომელთაც აქვთ ოპტიმალური თავისუფალი მანძილი, მაგრამ უარესი დისტანციური პროფილი, ვიდრე მე-7 ცხრილში მოცემულ კოდერებს. მე-7 და მე-8 ცხრილების კოდერები აგებული იყო რ. იოჰანესონისა და ე. პაასკეს მიერ [32].

ცხრილი 8

M	$G^{(1)}$	$G^{(2)}$	d_M	გზების რიცხვი წონით d_M	d_∞	გზების რიცხვი წონით d_∞
11	7173	5261	7	6	15	14
12	53734	72304	7	3	16	14
14	63121	55367	8	12	18	29
15	447254	627324	7	2	19	30
16	716502	514576	8	5	20	53

26. კოდების პრაქტიკული გამოყენება

შეცდომების კონტროლი წარმოადგენს მზარდი მნიშვნელობის მქონე პრობლემას თანამედროვე სატელეკომუნიკაციო სისტემებისათვის. ეს განპირობებულია იმით, რომ მონაცემთა ინტეგრირებას ენიჭება განსაკუთრებული მნიშვნელობა თანამედროვე ციფრული საზოგადოებისათვის. იზრდება მოთხოვნები გადაცემული მონაცემებში შეცდომის ალბათობაზე, როგორც საკომუნიკაციო, ისე მონაცემების

შენახვის სისტემებში. ამასთან მნიშვნელოვნად არის გაზრდილი მონაცემთა გამტარუნარიანობა და მოცულობები. გარკვეული მონაცემები არ შეიძლება იყოს არასწორი; მაგალითად, ვერავინ შეძლებს შეაფასოს იარაღის მაკონტროლებელ სისტემებზე მონაცემების დაუდგენელი შეცდომის გავლენა. ზოგადად, ნებისმიერ სისტემაში, რომელიც ამუშავებს მონაცემთა დიდ რაოდენობას, გაუსწორებელ და გაუმჟღავნებელ შეცდომებს შეუძლია გააუარესოს მახასიათებლები, რეაგირების დრო და გაზარდოს ადამიანური რესურსების ჩარევის საჭიროება. მნიშვნელოვანია იმის გაცნობიერება, რომ შეცდომების კონტროლი არის აპარატურის დიზაინის შემადგენელი ნაწილი, რომელსაც ძირეულად შეუძლია შეცვალოს საკომუნიკაციო სისტემის შემუშავების პრინციპები. მოვიყვანოთ რამდენიმე მაგალითი: სატელიტური კომუნიკაციების დროს, დაბალი სიჭარბის ეფექტურ კოდირებას შეუძლია შეამციროს გადამცემის სიმძლავრე, გაამარტივოს დედამიწის სადგურების აპარატურა და უზრუნველყოს გეოსტაციონალური სინქრონული სატელიტების ოპტიმალური განლაგება. არაკოდირებულ მოდულაციასთან შედარებით გისოსურად კოდირებული მოდულაცია უზრუნველყოფს მეტი მოცულობის მონაცემების გადაცემას შეზღუდული გამტარიანურობის მქონე არხებში, რაც ზრდის სპექტრულ და ენერგეტიკულ ეფექტურობას. ფლეშ-მეხსიერებაში არსებობს რამდენიმე სახის ხმაურის წყარო, როგორიცაა შემთხვევითი ხმაური, შეკავების პროცესი, უჯრედთაშორისი ჩარევა, ფონის ხმაური, ჩაწერა/წაკითხვის პროგრამის დარღვევა და ა. შ. ხმაურის ასეთი წყაროები მნიშვნელოვნად ამცირებს ფლეშ-მეხსიერების საიმედოობას. უკანას-

კნელ პერიოდში ხელშეშლამდგრადი კოდირების გამოყენება მნიშვნელოვნია ამ პრობლემების გადაჭრის პროცესში.

მიგვაჩნია, რომ სახელმძღვანელოში წარმოდგენილი კოდირების სქემები ქმნის ფუნდამენტს, სხვა უფრო რთული ხელშეშლამდგრადი სისტემების შესწავლისათვის, რომლებიც სადღეისოდ გამოიყენება ინფორმაციის გადამცემ და შემნახველ სისტემებში [33]-[34].

ლიტერატურა

1. M. Bossert. Channel Coding for Telecommunications. John Wiley & Sons, Ltd, Chichester, 1999.
2. S. Linand and D. Costello. Error Control Coding. Upper Saddle River, Prentice-Hall, NJ, 2004.
3. E. R. Berlekamp. Algebraic Coding Theory. Aegean Park Press, Laguna Hills, 1984.
4. W. W. Peterson and E. J. Weldon. Error-Correcting Codes. Cambndge, Massachusetts: MIT Press, 1972.
5. R. G. Gallager. Information Theory and Reliable Communication. New York: Wiley, 1968.
6. J. L. Massey "Error bounds for tree codes, trellis codes, and convolutional codes with encoding and decoding procedures," in Coding and Complexity, G. Longo, Ed. New York: Springer, 1976.
7. ნ. ხარატიშვილი. სიგნალების გადაცემის თეორია. „განათლება“, თბ., 1984.
8. ჯ. ბერიძე, ს. შავგულიძე. მეთოდური მითითებები საკურსო და სადიპლომო პროექტების შესასრულებლად დისკრეტული შეტყობინებების გადაცემაში. სპი-ს გამომცემლობა, თბ., 1986.
9. ს. შავგულიძე, ნ. უღრელიძე. ხვევადი კოდირების თეორიული საფუძვლები. დამხმარე სახელმძღვანელო. სტუ-ს გამომცემლობა, თბ., 1992.
10. R. Johannesson. On the error probability of general tree and trellis codes with applications to sequential decoding.

Tech. Rep. №EE7316, Dept. of Elec. Univ. of Notre Dame, Indiana, U.S.A. 1973.

11. G. D. Forney. Convoluttonal codes II. Maximum-likekihood decoding. Information and Control. 1974. Vol. 25. № 3. P. 222-266.
12. К. Ш. Зигангиров. Новые асимптотические нижние границы свободного расстояния для постоянных во времени сверточных кодов. Пробл. Передачи Информ. 1986. Т. 22, № 2. С. 34-42.
13. A. J. Viterbi. Convolutioanal codes and their parformance in communication systems. IEEE Trans. Comm. Tech. 1971. Vol. 19. № 5. P. 751-772.
14. L. Van de Meeberg. A tightened upper bound on the error probability of binary convolutional codes with Viterbi decoding. IEEE Trans. Inform. Theory. 1974. Vol. 20. № 3. P. 389-391.
15. K. J. Larsen. Short convolutional codes with maximm free distance for rates 1/2, 1/3 and 1/4. IEEE Trans. Inform. Theory. 1973. Vol. 19. № 3. P. 371-372.
16. R. Johannesson. Robustly optimal rate one-half binary convolutional codes. IEEE Trans. Inform. Theory. 1969. Vol. 15. № 5. P. 631-636.
17. G. S. Lauer. Some optimal partial unit memory codes. IEEE Trans. Inform. Theory. 1979. Vol. 25. № 2. P. 240-243.
18. G. D. Forney. Convolutional codes I. Algebraic structure. IEEE Trans. Inform. Theory. 1970. Vol. 16. № 6. P. 720-738.

19. J. L. Massey and M. K. Sain. Inverses of linear sequential circuits. IEEE Trans. Computers. 1968. Vol. 17. № 4. P. 330-337.
20. A. J. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. IEEE Trans. Inform. Theory. 1967. Vol. 13. № 2. P. 260-269.
21. J. K. Omura. On the Viterbi decoding algorithm. IEEE Trans. Inform. Theory. 1969. Vol. 15. № 1. P. 177-179.
22. Consultative Committee for space Data Systems, “Recommendation for Space Data Systems Standard, Telemetry channel coding”, CCSDS 101. O-B-2, blue book, Issue 2, Jan. 1987.
23. R. M. Fano. A heuristic discussion of probabilistic decoding. IEEE Trans. Inform. Theory. 1963. Vol. 9. № 2. P. 64-74.
24. J. L. Massey. Variable-Iength codes and the Fano metric. IEEE Trans. Inform. Theory. 1972. Vol. 18. № 1. P. 196-198.
25. К. Ш. Зигангиров. Некоторые последовательные процедуры декодирования. Пробл. Передачи Информ. 1966. Т. 2, № 4. С. 13-15.
26. F. Jelinek. A fast sequential decoding algorithm using a stack. IBM J. Res. Dev. 1969. Vol. 13. № 11. P. 675-685.
27. Дж. Возенкрафт, И. Джекобс. Теоретические основы Техники связи. М., Мир, 1969.
28. J. M. Geist. Search properties of some sequential decoding algorithms. IEEE Trans. Inform. Theory. 1973. Vol. 19. № 4. P. 519-526.

29. J. M. Geist. An empirical comparison of two sequential decoding algorithms. IEEE Trans. Comm. Tech. 1971. Vol. 19. № 8. P. 519-526.
30. E. Paaske. Short binary convolutional codes with maximal free distance for rates 2/3 and 3/4. IEEE Trans. Inform. Theory. 1973. Vol. 20. № 5. P. 683-689.
31. Y. Yasuda, K. Kashiki, and Y. Hirata. High-rate punctured convolutional codes for solt decision Viterbi decoding. IEEE Trans. Comm. Tech. 1984. Vol. 32. № 3. P. 315-319.
32. R. Johannesson and E. Paaske. Further results on binary convolutional codes with an optimum distance profile. IEEE Trans. Inform. Theory. 1978. Vol. 24. № 2. P. 264-268.
33. ნ. უღრელიძე, ს. შავგულიძე. ციფრული კავშირის სისტემების შემუშავება რადიოარხებისათვის. კავკასიის უნივერსიტეტის გამომცემლობა, თბ., 2020.
34. ს. შავგულიძე, ნ. უღრელიძე. მეექვსე თაობის (6G) უსადენო საკომუნიკაციო ქსელები და სისტემები. კავკასიის უნივერსიტეტის გამომცემლობა, თბ., 2021.

სარჩევი