

УДК 681.3

А.Я. КУЛИК, С.Г. КРИВОГУБЧЕНКО, Ю.Ю. ІВАНОВ

## ТУРБО-КОД: ВЛАСТИВОСТІ ТА ЗАСТОСУВАННЯ ДЛЯ ЗАВАДОСТІЙКОГО КОДУВАННЯ

*Вінницький національний технічний університет  
95, Хмельницьке шосе, м. Вінниця, 21021, Україна  
Тел. (+380) (0432)598 437. E-mail: kulyk@inaeksu.vstu.vinnica.ua*

**Анотація.** Розглядаються питання, пов'язані з сучасним турбо-кодуванням, що є найперспективнішим для застосування у волоконно-оптичних лініях зв'язку. На основі простого кодера за допомогою алгоритма Вітербі та Trellis-діафрагми показана процедура кодування. Розглядаються існуючі методи декодування. Описується метод SOVA, побудований на основі прогнозу апостеріорної ймовірності кожного окремого біта.

**Аннотация.** Рассматриваются вопросы, связанные с современным турбо-кодированием, которое является самым перспективным для применения в волоконно-оптических линиях связи. На основе простого кодера с помощью алгоритма Витерби и Trellis-диафрагмы показана процедура кодирования. Рассматриваются существующие методы декодирования. Описывается метод SOVA, который построен на основе прогноза апостериорной вероятности каждого отдельного бита.

**Abstract.** The questions, connected with a modern turbo-coding, which is the most perspective for application in fiber-optical communication lines, are considered. On the basis of the simple coder by means of algorithm of Viterbi and Trellis-diaphragm is shown the coding procedure. Existing methods of decoding are considered. Method SOVA, which is constructed on the basis of the forecast of a posteriori probability of every separate bit, is described.

**Ключові слова:** завадостійке кодування, турбо-код, дібіт, Trellis-кодування, алгоритм Вітербі, SOVA, SISO.

### ВСТУП

Для боротьби із завадами в системах передавання даних застосовується завадостійке кодування. Воно дозволяє здійснити захист даних в пам'яті обчислювального пристрою та накопичувачах інформації; використовується для стиснення інформації.

Турбо-код, розроблений в 1993 р. та описаний в наукових працях С. Berrou, А. Glavieux, Р. Titimajshima, досить потужний метод виправлення помилок, що має важливу перевагу в тому, що дозволяє забезпечити надійний зв'язок з ефективністю дуже близькою до теоретичної границі Клода Шенона. Турбо-коди є новим класом методів, що визначають ітеративне кодування з виправленням помилок. Розглянутий код є блоковим, але будується на основі згорткових кодів. Використовуватись може в будь-якій системі зв'язку. Турбо-код удосконалюється у напрямі підвищення швидкості та надійності передавання даних, що безперечно робить його застосування найбільш перспективним в волоконно-оптичних лініях зв'язку. Одними з найважливіших сфер використання такого кодування на практиці є волоконно-оптичні системи зв'язку, цифрове телебачення, космічні та мобільні супутникові комунікації [1, 2]. Отже, турбо-коди застосовують, щоб кодувати об'ємні інформаційні повідомлення для високошвидкісного передавання даних з великою завадостійкістю.

Актуальною є проблема реалізації розроблених обчислювальних процедур в техніці, приймання та формування турбо-кода з використанням обчислювальної потужності технічних пристроїв, а також дослідження ймовірнісно-енергетичних характеристик передачі інформації з використанням технічних засобів.

Використання турбо-кодування дозволяє працювати при значно більш низьких відношеннях сигнал/шум, ніж при застосуванні згорткового кодування по Вітербі або каскадних кодів Вітербі-Ріда-Соломона. Розглянутий код забезпечує роботу при співвідношенні сигнал/шум приблизно 2 дБ в лініях, обмежених по доступній потужності [3].

СТРУКТУРА ТУРБО-КОВОДОЇ СИСТЕМИ

Для турбо-ководової системи (рис. 1) зі швидкістю  $\frac{1}{3}$  [4, 5] доцільно використати верхній регістр для позначення двійкових чисел, а нижній регістр – для значень сигнал/символ.

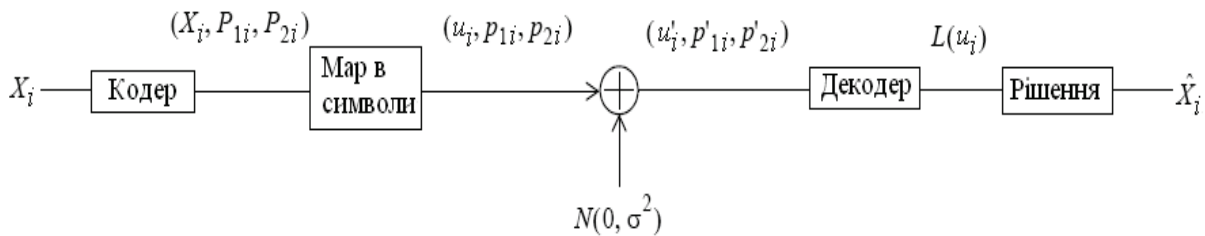


Рис. 1. Система турбо-кода:

$X_i$  – подана послідовність інформаційних бітів;  $P_{1i}$  – дібіт з кодера 1;  $P_{2i}$  – дібіт з кодера 2;  $u_i$  – інформаційні символні дані;  $p_{1i}$  – символи парності з декодера 1;  $p_{2i}$  – символи парності з декодера 2;  $\sigma^2$  – дисперсія шуму;  $u'_i$  – зашумовані інформаційні символні дані;  $p'_{1k}$  – зашумовані символи парності з декодера 1;  $p'_{2k}$  – зашумовані символи парності з декодера 2;  $L(u_i)$  – “м’яке” рішення декодера;  $\hat{X}_i$  – “жорстке” рішення декодера

Схема турбо-кодера з двома згортковими ідентичними кодерами та перемежувачем  $P$  (permuter, interleaver) наведена на рис. 2.

Завданням перемежувача є перетворення вхідної інформаційної послідовності таким чином, щоб комбінації, що призводять до кодових слів з низькою вагою (вагою називається число ненульових бітів кодового слова) на виході першого кодера, були перетворені в комбінації, що породжують кодові слова з високою вагою на виході інших кодерів, тим самим забезпечуючи невелике число кодових слів малої ваги результуючого турбо-кода. Перемежувач є одним із способів боротьби з помилками.



Рис. 2. Схема турбо-кодера

Структура турбо-декодера (рис. 3) складається з пари декодерів, які працюють спільно для того, щоб уточнити і покращити оцінку початкових інформаційних бітів.

Декодери працюють на основі алгоритму декодування по максимуму апостеріорної імовірності і випускають інформацію “м’якого” рішення. Спочатку декодер № 1 починає роботу без ініціалізації інформації (апріорні оцінки дорівнюють нулю). У наступних ітераціях інформація “м’яких” рішень використовується для активації іншого декодера. Інформація з декодера обертається петлею до того моменту, поки “м’які” рішення не сходяться на стабільному наборі значень. Останні “м’які” рішення використовуються для відновлення вихідної бінарної послідовності.

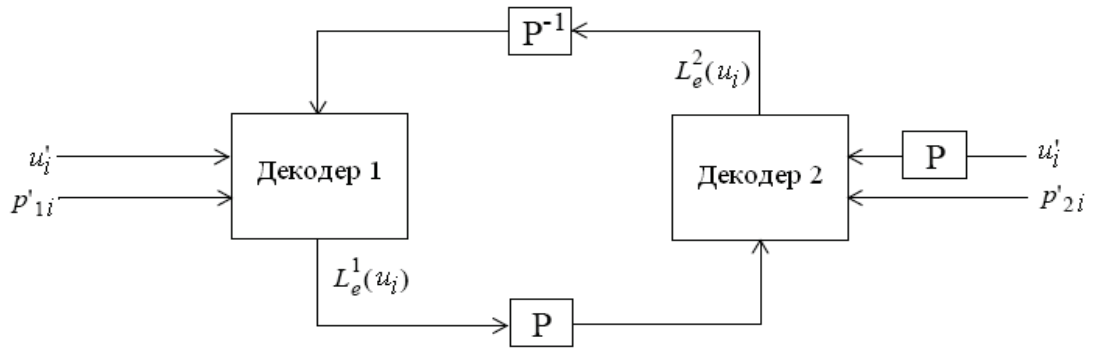


Рис. 3. Схема турбо-декодера

### МЕТОДИ КОДУВАННЯ ТА ДЕКОДУВАННЯ ТУРБО-КОДІВ

При роботі з турбо-кодами формується діафрагма у вигляді решітки (Trellis coding). Для найпростішого кодера (рис. 4), що складається з двох комірок пам'яті та елементів XOR, можна розглянути принцип формування дібітів [6].

Значення кожного дібіта, що формується, залежить не лише від вхідного інформаційного біта, але і від двох попередніх бітів, значення яких зберігається в комірках пам'яті. Якщо прийняти, що  $A_i$  – вхідний біт, то значення елемента XOR2 виражатиметься у виразі

$$A = A_i \oplus A_{i-1} \oplus A_{i-2}. \quad (1)$$

Значення елемента XOR3 можна подати у вигляді формули

$$B = A_i \oplus A_{i-2}. \quad (2)$$

Отже, можна побачити залежність значення дібіта від трьох факторів: значення вхідного біта та двох комірок пам'яті. Такі кодери отримали назву згорткових на три стани ( $K = 3$ ) з вихідною швидкістю 0,5.

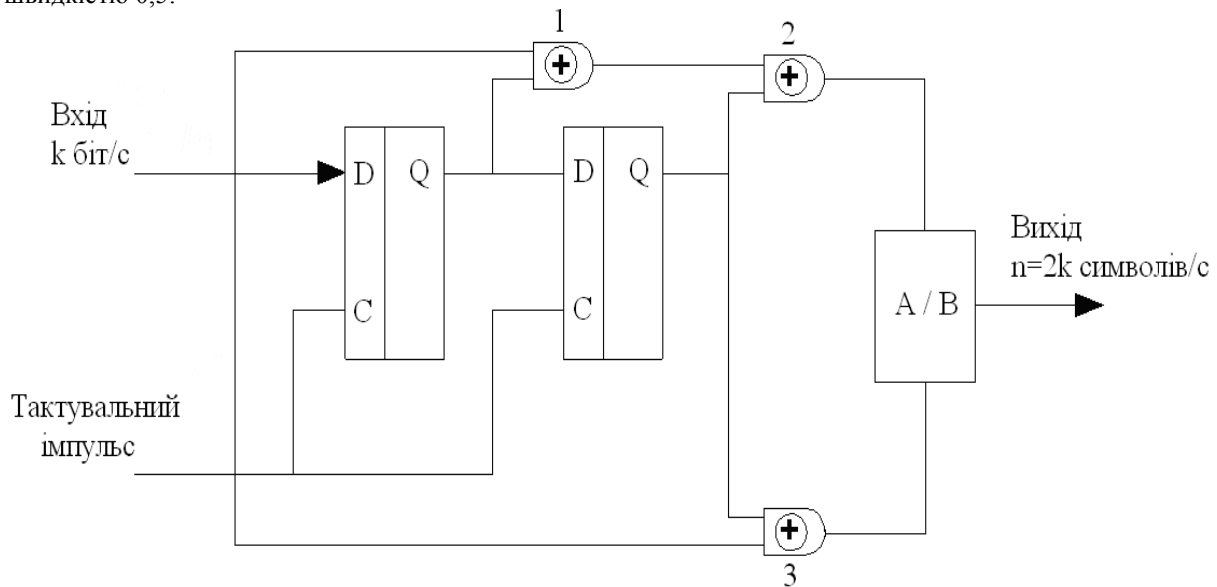


Рис. 4. Схема простого Trellis-кодера

Роботу кодера зручно розглядати не на основі часових діаграм, а використовуючи певні діаграми станів. Стани кодера визначаються за допомогою значень комірок пам'яті. Всього можливо чотири різних стани кодера: 00, 01, 10, 11.

Нехай в деякий момент часу стан кодера буде 00. Цікавим буде стан кодера в наступний момент часу та який дібіт буде при цьому сформований. На основі формул (1) та (2), проаналізувавши всі

можливі переходи із станів можна побудувати орієнтований граф на основі станів кодера або повну діаграму станів кодера (рис. 5), де в кружках визначаються стани кодера, а вхідний біт і дібїт, що формуються пишуться через косу лінію. На основі вище описаного обчислюється значення дібїта, що формуються кодером.

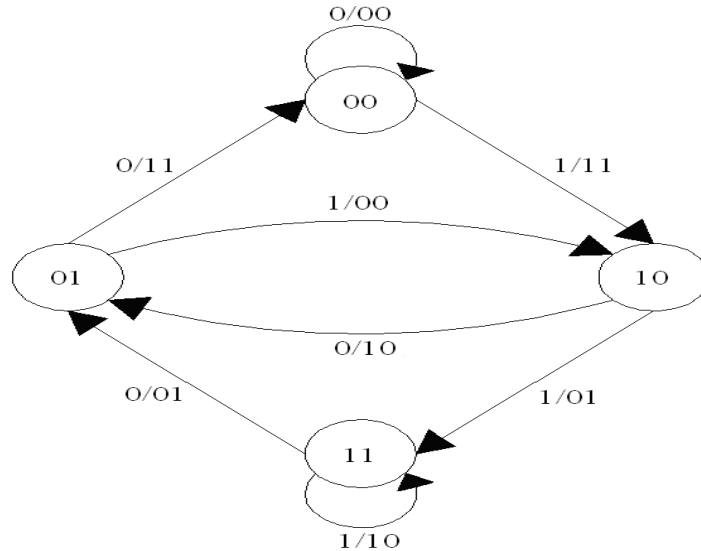


Рис. 5. Повна діаграма станів кодера

Використовуючи діаграму станів, можна побудувати часову діаграму переходів для послідовності бітів. Для цього будується таблиця, схожа на решітку (Trellis). Основною перевагою Trellis-форми (рис. 6) є завадостійкість. Оскільки, внаслідок надлишковості кодування (кожному інформаційному біту ставиться у відповідність дібїт, тобто надлишковість кода рівна 2), у випадку появи помилок приймання, вихідна послідовність може бути безпомилково відновлена. Для цього використовується декодер Вітербі. На рис. 6 напрямні стрілки показують перехід, що відповідає розглянутій комбінації бітів.

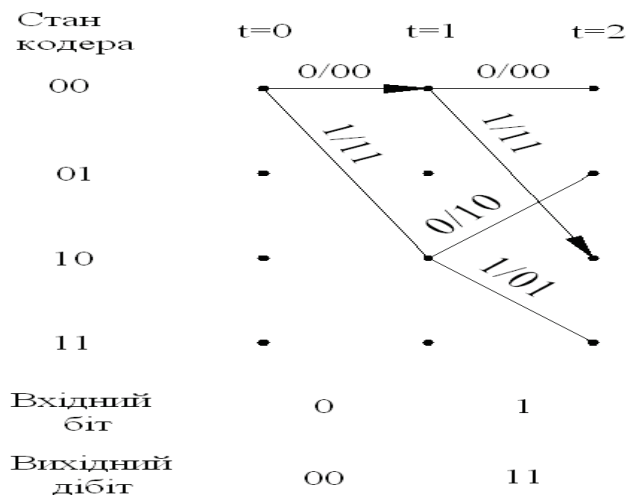


Рис. 6. Часова діаграма кодера для перших трьох моментів часу

Декодер Вітербі у випадку безпомилкового приймання всієї послідовності дібїтів буде мати інформацію про цю послідовність, а також про будову кодера (тобто діаграму його станів) та його початковий стан. Маючи всю цю інформацію, він повинен відновити послідовність бітів, що була спочатку. Для цього використовується відповідна таблиця та обчислюється метрика помилок. Для переходу, що відповідає прийнятому дібїту (тобто перехід правильний), метрика помилок приймається

рівною нулю, а для інших переходів вона розраховується за кількістю відмінних бітів в прийнятому дібіті та тому дібіті, який відповідає розглянутому переходу. Наприклад, якщо прийнятий дібіт 00, а дібіт, що відповідає переходу, рівний 11, то метрика помилок для даного переходу рівна 2. Отже, будується решітка на якій прокладається шлях, що має найкоротшу метрику накопичених помилок за всі такти кодера. Для цього зручно побудувати таблицю, в якій відображати всі метрики помилок для переходів із одного стану в інший за певний момент часу. Ті шляхи, які мають найменшу метрику помилок, називають “живими”.

Метод декодування за алгоритмом Вітербі являє собою декодування по максимуму правдоподібності. Ідея алгоритму полягає в тому, що в декодері відтворюються всі можливі шляхи послідовних змін станів сигналу за допомогою зіставлення отриманих при цьому кодових символів з прийнятими аналогами по каналу зв'язку. На основі аналізу помилок між прийнятими та необхідними символами визначають оптимальний шлях (оптимальною вважається та послідовність, відстань Хемінга якої від прийнятої послідовності мінімальна). Таким чином, проводиться пошук найвигіднішого максимально правдоподібного шляху на графі – діаграма кода у формі Trellis [7, 8].

Можна вважати, що лінійний блоковий код побудований за допомогою перетворення двійкового згорткового коду пам'яті  $m$  та швидкості  $\frac{1}{n}$ , а сигнали передаються по каналу з адитивним білим гаусівським шумом (Additive White Gaussian Noise – AWGN). Доцільно також припустити, що згортковий кодер починає роботу з нульового стану і через  $N$  Trellis-секцій (кроків) зупиняється в нульовому стані, чого завжди можна досягти в реальних умовах.

Декодування блокового коду можливе з використанням “жорсткого” (hard decision decoding – HDD) або “м'якого” (soft decision decoding – SDD) прийняття рішення, а на виході з декодера отримуються “жорсткі” або “м'які” дані. В HDD кожному прийнятому біту в демодуляторі приписується значення 0 або 1, в залежності від того, чи є прийняті дані із завадами більше або менше порогового значення.

Декодер з “м'яким” прийняттям рішення приймає крім бінарної величини 1 або 0 також довірчу величину, що пов'язана із заданим бітом. Якщо модулятор визначився, то біту присвоюється значення 1, ступінь упевненості в ньому висока. Якщо він менш визначений, то отримає низьку довірчу величину.

Декодер з “м'яким” входом може випускати дані з “жорстким” рішенням або дані з “м'яким” рішенням. Наприклад, декодуючий пристрій Вітербі приймає “м'яку” інформацію з демодулятора і випускає дані з “жорстким” рішенням.

Декодер з “м'яким” входом і “м'яким” виходом (Soft-Input Soft-Output – SISO) приймає дані з “м'яким” рішенням і випускає дані з “м'яким” рішенням.

Концепція декодерів SISO застосована в турбо-кодах. Турбо-код подає дані, що демодулюються, з “м'яким” рішенням в SISO-декодер. Вихідні дані цього декодера потім подаються в той же самий (чи інший) SISO-декодер. Потім операція повторюється знову. Цей ітераційний процес триває до тих пір, поки не буде прийняте впевнене рішення. Концепція подачі вихідних даних назад на вхід аналогічна турбонадуву двигуна, тому турбо-код дістав саме таку назву. Для ефективності застосування турбо-кода наявні дані слід закодувати двома (чи більше) різними кодами.

Для декодування турбо-кодів можна використовувати декілька методів: алгоритм Вітербі з “м'яким” виходом (soft-output Viterbi algorithm – SOVA); алгоритм декодування по максимуму апостеріорної імовірності (maximum a posteriori probability – MAP) та його модифікації для зниження обчислювальної складності (Log-MAP, Max-Log-MAP алгоритми) [7, 9].

Розглянемо категорію м'яких алгоритмів, що включає максимальну правдоподібність алгоритма, яка мінімізує ймовірність помилки на біт інформації за рахунок прогнозу апостеріорної ймовірності кожного окремого біта. Це алгоритм Вітербі з “м'яким” виходом (SOVA) [10]. Дана модифікація алгоритма Вітербі, розроблена в 1989 р., дозволяє обчислити надійність інформаційних символів як логарифм відношення правдоподібності (log-likelihood ratio – LLR), що можна записати у формулі

$$L(u_i) = \log \left( \frac{\text{Pr obability}(u_i = 1 | u')}{\text{Pr obability}(u_i = 0 | u')} \right), \quad (3)$$

де  $u'$  - прийнята послідовність.

Робота декодера SOVA ділиться на дві частини [11, 12]. У першій частині декодування відбувається так само, як у звичайному алгоритмі Вітербі, який вибирає найбільш імовірну кодову

послідовність, що відповідає шляху прокладеному на Trellis-діаграмі “вперед”. Але потрібно зберегти всі метрики на кожному кроці і для кожного стану декодера. У другій частині SOVA алгоритм Вітербі виконується “назад”, зберігаючи метрики для кожного стану решітки. Після другого проходу для кожного вузла решітки формуються дві метрики.

Через  $Y_{\max}$  позначається метрика найбільш правдоподібної послідовності  $\omega$ , знайденої алгоритмом Вітербі. Апостеріорна ймовірність (APP) залежить від  $Y_{\max}$  так як справедлива наступна оцінка

$$\Pr obability\{u | u'\} = \Pr obability\{\omega | u'\} \approx e^{Y_{\max}}. \quad (4)$$

Нехай  $Y_i(1) \approx Y_{\max}$ , тоді справедлива формула

$$\Pr obability\{u_i = 1 | u'\} = e^{Y_i(1)}. \quad (5)$$

На стадії формування “м’якого” виходу алгоритма SOVA для  $i$ -ої секції визначається найбільш правдоподібне значення інформаційного символу  $u_i = a$ , де  $a \in \{0, 1\}$ , і відповідна йому максимальна метрика, що знайдена на прямому проході Вітербі, встановлюється рівною  $Y_i(u_i)$ . Метрика найкращого шляху для доповнення  $Y_i(u_i \oplus 1)$  може бути знайденою за формулою

$$Y_i(\omega_i \oplus 1) = \min_{k_1, k_2} \{Y_f(S_{i-1}^{(k_1)}) + BY_i^{(b_1)}(u_i \oplus 1) + Y_b(S_i^{(k_2)})\}, \quad (6)$$

де  $k_1, k_2 \in \{0, 1, 2, \dots, 2^n - 1\}$ ;  $Y_f(S_{i-1}^{(k_1)})$  – метрика шляху, що “вижив” на прямому проході за алгоритмом Вітербі для  $(i-1)$ -ої секції и стану  $S^{(k_1)}$ ;  $BY_i^{(b_1)}(u_i \oplus 1)$  – метрика ребра для інвертованого інформаційного символу, асоційованого з переходом зі стану  $S^{(k_1)}$  в  $S^{(k_2)}$ ;  $Y_b(S_i^{(k_2)})$  – метрика шляху, що “вижив” на зворотньому проході за алгоритмом Вітербі для  $i$ -го моменту та стану  $S^{(k_2)}$ .

Якщо позначити через  $Y_i(0)$  максимальну метрику шляху, асоційованого з доповненням інформаційного символу  $u_i$ , то можна навести формулу для обчислення м’якого виходу.

$$L(u_i) = Y_i(1) - Y_i(0). \quad (7)$$

Наприклад, задано код С (12, 4), побудований зі згорткового кода з пам’яттю 2, швидкістю 0,5 та з генераторами (7, 5). Можна вважати, що інформаційна послідовність має вигляд  $u = (110100)$ , а прийнята двійкова послідовність на виході каналу AWGN виглядає

$$u' = (-4, -1, -1, -3, +2, -3, +3, +3, -3, +3, -3, +1).$$

На рис. 7 показана решітка цього коду. Для всіх  $i = 0, 1, \dots, 6$  над кожним станом решітки

написана мітка у вигляді  $\frac{Y_f(S_i^{(m)})}{Y_b(S_i^{(m)})}$ . Ребра решітки позначені метрикою  $BY_i$ . Тоді на основі поданого

вище та [12] отримаємо, що м'які виходи  $L(u_i)$  для  $i = 0, 1, \dots, 6$  рівні, відповідно, -34, -16, +22, -16, +24, +24.

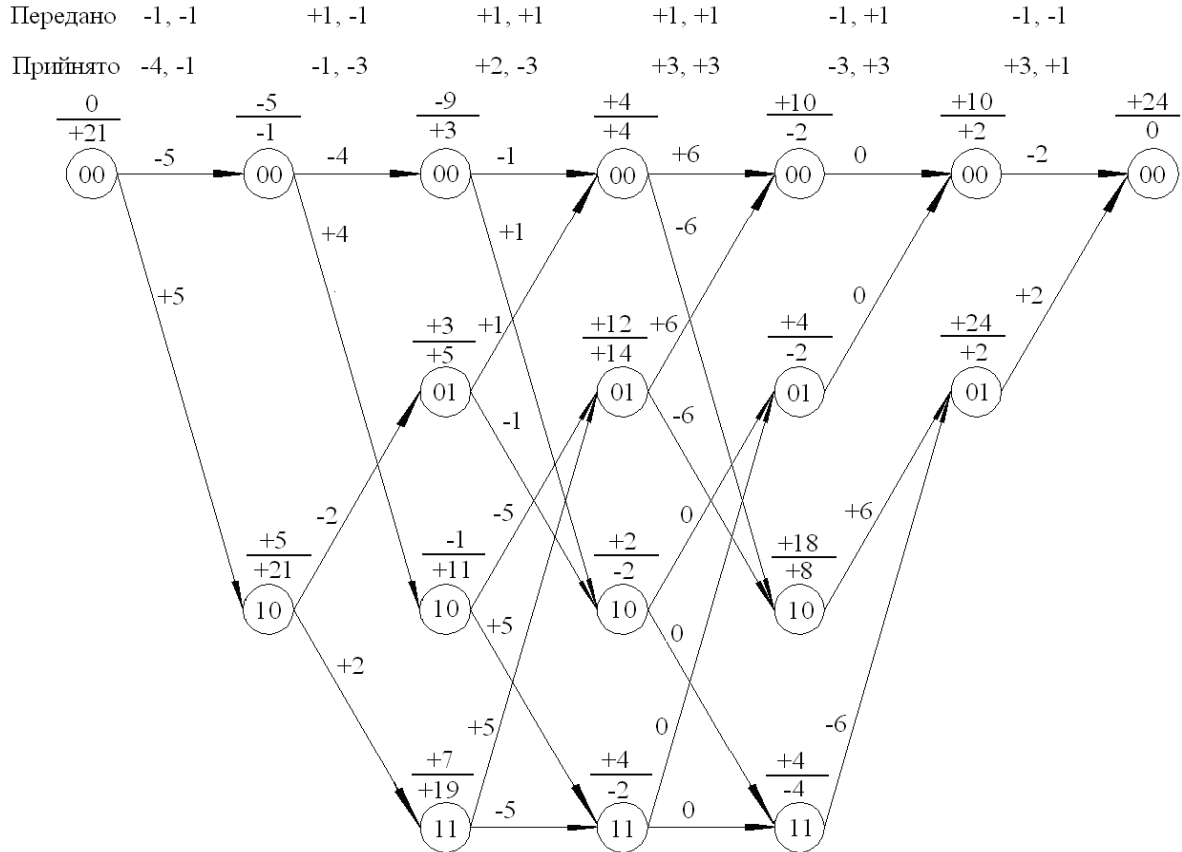


Рис. 7. Trellis-діафрагма, використана для SOVA декодування

### ВИСНОВКИ

Дослідження систем із застосуванням турбо-кодів актуальні. Таким чином, очікується, що турбо-коди стануть стандартом для кодування техніки цього тисячоліття.

### СПИСОК ЛІТЕРАТУРИ

1. Breiling M. The Super-Trellis Structure of Turbo Codes / M. Breiling, L. Hanzo // IEEE Transactions on Information Theory. – 2000. – V. 46. – № 6. – P. 2212-2228.
2. Скляр Б. Цифровая связь. Теоретические основы и практическое применение: пер. с англ.; под ред. А.В. Назаренко / Б. Скляр. – М.: Издательский дом «Вильямс», 2003. – 1004 с. – ISBN: 5-8459-0497-8.
3. Диденко М.Г. Радиосигналы в спутниковых системах связи [Электронный ресурс] / М.Г. Диденко. – Режим доступа: [http://www.vsat-tel.ru/library/art\\_43.htm](http://www.vsat-tel.ru/library/art_43.htm).
4. Ryan W. A Turbo Code Tutorial [Web resource] / W. Ryan. – Access mode: <http://vada.skku.ac.kr/ClassInfo/digital-com2000/slides/turbo2c.pdf>.
5. Berrou C. Near Shannon Limit Error-Correcting Coding and Decoding: Turbo-Codes / C. Berrou, A. Glavieux, P. Thitimajshima // IEEE Transactions on Information Theory. – 1996. – V. 44. – № 10. – P. 1064-1070.
6. Пахомов С. Технология 1000Base-T на физическом уровне [Электронный ресурс] / С. Пахомов. – Режим доступа: <http://www.compress.ru/article.aspx?id=9774&iid=412#begin>.

7. Lesson 35. Convolutional Codes [*Web resource*] : [*Webcourse*] — Access mode: <http://nptel.iitm.ac.in/courses/Webcourse-contents/ИТ%20Kharagpur/Digi%20Comm/pdf-m-6/m6l35.pdf>.
8. Блейхут Р. Теория и практика кодов, контролирующих ошибки: пер. с англ.; под ред. И.И.Грушко / Р. Блейхут. — М.: Мир, 1986. — 576 с.
9. Витерби А.Д. Принципы цифровой связи и кодирования: пер. с англ.; под ред. К.Ш. Зигангирова / А.Д. Витерби, Дж.К. Омура. — М.: Радио и связь, 1982. — 536 с.
10. Viterbi J. Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm / J. Viterbi // IEEE Transactions on Information Theory. — 1967. — V. IT-13. — P. 260-269.
11. Кларк Дж. Кодирование с исправлением ошибок в системах цифровой связи: пер. с англ.; под ред. Б.С. Цыбакова / Дж. Кларк, Дж. Кейн. — М.: Радио и связь, 1987. — 382 с.
12. Морелос-Сарагоса Р. Искусство помехоустойчивого кодирования. Методы, алгоритмы, применение: пер. с англ.; под ред. В.Б. Афанасьева / Р. Морелос-Сарагоса. — М.: Техносфера, 2005. — 320 с. — ISBN: 5-94836-035-0.

Надійшла до редакції 21.10.2011р.

**КУЛИК АНАТОЛІЙ ЯРОСЛАВОВИЧ** – д.т.н., професор кафедри автоматики та інформаційно-вимірювальної техніки, Вінницький національний технічний університет, Вінниця, Україна.

**КРИВОГУБЧЕНКО СЕРГІЙ ГРИГОРОВИЧ** – к.т.н., доцент кафедри автоматики та інформаційно-вимірювальної техніки, Вінницький національний технічний університет, Вінниця, Україна.

**ІВАНОВ ЮРІЙ ЮРІЙОВИЧ** – студент факультету автоматики та комп'ютерних систем управління ІнАЕКСУ, Вінницький національний технічний університет, Вінниця, Україна.