УДК 004.056.2

О. Я. Матов[1], В. С. Василенко[2], М. Ю. Василенко[2]
[1]Інститут проблем реєстрації інформації НАН України
вул. М. Шпака, 2, 03113 Київ, Україна
[2]Національний авіаційний університет
вул. Космонавта Комарова, 1, 03058 Київ, Україна

Контроль і поновлення цілісності на основі алгоритму нулізації у коді умовних лишків

Запропоновано процедуру виявлення місця та величини спотворень інформаційних об'єктів телекомунікаційних мереж на основі алгоритму нулізації в умовах застосування узагальненого коду умовних лишків.

Ключові слова: *завадостійке кодування, код умовних лишків, контроль цілісності, контрольна основа, основа коду, поновлення цілісності, спотворення.*

Вступ

Під цілісністю інформації у статті будемо розуміти відсутність у ній будь-яких спотворень (модифікацій), які не були санкціоновані її власником, незалежно від причин або джерел виникнення таких спотворень. Спотворення інформації, тобто порушення її цілісності, можливі на будь-якому етапі її циркуляції в обчислювальних мережах: при зберіганні, передачі або обробці. Використання спотвореної інформації здатне викликати наслідки (часто надзвичайно важкі) для власників або користувачів цієї інформації. Тому задача забезпечення цілісності та доступності інформаційних ресурсів є однією з найактуальніших при розробці й експлуатації автоматизованих систем і їхніх елементів.

Для забезпечення контролю та поновлення цілісності інформаційних об'єктів, включаючи й відновлення зруйнованої інформації, до складу інформації, яка захищається, включають надмірну інформацію — ознаку цілісності або контрольну ознаку (залежно від прийнятої у задачах контролю цілісності або завадостійкого кодування термінології).

При цьому між інформацією, що захищається, і ознаками цілісності встановлюється регулярний (функціональний) односторонній зв'язок (процедури розрахунку контрольної ознаки за початковою інформацією, що захищається, відомі, а процедури розрахунку початкової інформації за контрольними ознаками найчастіше не існує). Контроль цілісності інформації (на відсутність спотворень), а в по-

дальшому і її відновлення, ґрунтуються при цьому на тих або інших процедурах перевірки наявності та характеру вказаного регулярного одностороннього зв'язку між ознаками цілісності та прийнятою інформацією. Ці процедури визначаються застосованим завадостійким кодом.

Отже основними процедурами завадостійких кодів є: процедура кодування (розрахунок контрольної ознаки) та декодування — перевірка наявності зв'язку між ознаками цілісності і прийнятою інформацією. При кодуванні по вихідній інформації створюється контрольна ознака (бажано мінімальної надлишковості) у вигляді своєрідного образу відображення цієї інформації, процедура формування якого відома, і який з дуже високою вірогідністю відповідає інформації, що захищається. При декодуванні здійснюється у тій чи іншій формі формування нової контрольної ознаки та її порівняння із попередньою. При спотворенні результати порівняння є негативними. Цей факт дає змогу стверджувати про наявність спотворень (цього досить у завадостійких кодах без корекції спотворень), а ступінь відмінності (у завадостійких кодах з корекцією спотворень) — про можливі місце та величину викривлень.

Поруч із задачею забезпечення цілісності інформаційних об'єктів у сучасних телекомунікаціях не менш важливою є задача забезпечення доступності цих об'єктів. З цією метою прагнуть підвищувати швидкість інформаційного обміну. Однією із можливостей цього є застосування багаторівневих (багатопозиційних) символів. Однак при цьому викривлення одного із символів призводить до появи групового (у декількох бітах) спотворення (спотворення узагальненого символу). Для виявлення таких спотворень доцільно застосовувати спеціальні коди для виявлення і (можливого) виправлення спотворень в узагальнених символах. Одним із таких кодів є код умовних лишків (ЛУ-коду) [1, 2].

Відомі алгоритми кодування-декодування в таких кодах. Одним із таких алгоритмів є, так званий, *z*-алгоритм. Його перевагою є відносна простота суто алгебраїчних процедур, але суттєвим недоліком — застосування в цих процедурах операції обчислення цілої частини від нескінчених дробових чисел, що може стати причиною неправильного кодування-декодування. Іншим є алгоритм нулізації, який не має операцій над нескінченими дробовими числами, але у відомих [3] описах і реалізаціях цей алгоритм використовується лише для виявлення факту наявності викривлень.

Метою цієї роботи є розроблення процедури застосування алгоритму нулізації не лише в режимі виявлення, а й у режимі визначення характеру та корегування спотворень.

Використання алгоритму нулізації для виявлення спотворень

Як відомо [3], внаслідок декодування із застосуванням процедури нулізації отримується величина, яка має по всім основам, окрім контрольної, лишки, що дорівнюють нулю, а по контрольній — лишок, величина якого

$$\gamma = (kP) \pmod{p_k},$$

тобто

$$\gamma = (0, 0, \ldots, 0, \ldots 0, (k \cdot P) \pmod{p_k}),$$

де $k = 0, 1, 2, \ldots, p_k - 1$. Для неспотворених чисел, тобто при $\gamma = 0$, величина $k = 0$, для спотворених — $\gamma \neq 0$. Отже, *визначений алгоритм надає змогу виявлення факту наявності спотворень.*

Для подальшої ілюстрації можливостей алгоритму розглянемо приклади кодування та декодування при виявленні наявності спотворень.

Приклад 1. Нехай необхідно закодувати з використанням алгоритму нулізації вихідний код 110110, вважаючи, що довжина узагальненого символу, а отже і можлива довжина пакету викривлень $b = 2$. Тоді можливе розбиття вихідного коду на три ($n = 3$) двохрозрядні групи $\alpha_1 = 11_2$, $\alpha_2 = 01_2$, $\alpha_3 = 10_2$, $s = 4$, а як умовні основи можна вибрати $p_1 = 4$, $p_2 = 5$, $p_3 = 7$. При цьому як значення контрольної основи можна вибрати $p_к = 71$ (нагадаємо, що контрольна основа повинна задовольняти умові $p_к > 2 \cdot p_n \cdot p_{n-1} = 2 \cdot 5 \cdot 7 = 70$), що потребує для свого відображення семи розрядів. Унаслідок цього для кодування формується код

$$A = 11.01.10.0000000.$$

Перше мінімальне число t_1 повинно мати лишок по першій основі, що дорівнює $11_{(2)} = 3_{(10)}$. Таким числом, є $t_1 = 3$, або при представленні в ЛУ-коді з вибраними основами

$$t_1 = 11.11.011.0000011.$$

Друге мінімальне число t_2 повинно мати лишок по першій основі, який дорівнює нулю, а по другій —

$$((\alpha_2 - \alpha_2^1) \pmod{p_2}) = (1 - 3) \pmod 5 = 11_{(2)}.$$

Мінімальним числом, яке має такі лишки по першій і другій основам, є $t_2 = 8$, тобто

$$t_2 = 00.11.001.0001000.$$

Третє мінімальне число t_3 повинно мати нульові лишки по першим двом основам, а по третій —

$$((\alpha_3 - \alpha_3^1 - \alpha_3^2) \pmod{p_3}) = (2 - 3 - 1) \pmod 7 = 5 = 101_{(2)}.$$

Мінімальним числом, що має такі лишки, є $t_3 = 40$, тобто

$$t_3 = 00.00.101.0101000.$$

Тоді сума цих чисел $T = \sum_{i=1}^{3} t_i$ дорівнює 51, тобто

$$T = 11.01.10.0110011.$$

Код $T = A$ є результатом кодування.

Приклад 2. Декодувати з використанням алгоритму нулізації базове кодове слово $\tilde{A}$ = 11.01.01.0110011, в якому спотворена третя пара розрядів. Як і раніше

$$t_1 = 11.11.011.0000011,$$
$$t_2 = 00.11.001.0001000.$$

Для третього мінімального числа t_3

$$((\alpha_3 - \alpha_3^1 - \alpha_3^2) \pmod{p_3}) = (1 - 3 - 1) \pmod 7 = 4 = 100_{(2)}.$$

Мінімальним числом, що має такі лишки, є $t_3 = 60$, тобто

$$t_3 = 00.00.100.\ 0111100.$$

При цьому

$$T = \sum_{i=1}^{3} t_i = 71,$$

але оскільки $T(\text{mod } 71) = 71(\text{mod } 71) = 0$, то

$$T = 11.01.01.0000000$$

і

$$\gamma = (\alpha_k - (T \bmod p_к)) \pmod{p_к} = (0110011 - 0000000) \pmod{71} = 51.$$

Оскільки $\gamma \neq 0$, то можна зробити *висновок про наявність спотворення в числі, що декодується*.

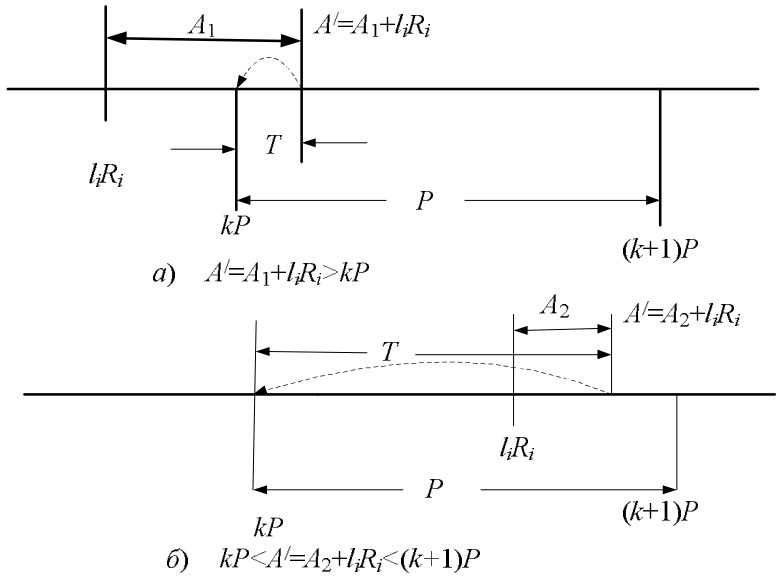
Використання алгоритму нулізації для корегування спотворень

Надалі покажемо, що за результатами нулізації можна визначити не лише факт наявності спотворень, але й їхнє місце та величину. Одним із можливих кроків на цьому шляху є визначення, до якого з інтервалів (піддіапазонів) величиною P потрапляє спотворене число.

Нагадаємо, що введення контрольної основи величиною p_k призводить до розширення діапазону представлення P в p_k разів, іншими словами, утворює p_k піддіапазонів величиною P кожен. Нагадаємо також [3], що спотворене число може бути представленим як сума початкового (неспотвореного) числа A та вектора

спотворень E: $\widetilde{A} = A + E$, де вектор спотворення E у системі лишкових класів (СЛК) має лишки, що дорівнюють нулю, по усім основам, окрім тієї, де є спотворення. Тобто вектор спотворення є числом вигляду $E = 0,0,...,\Delta A,0,0,...,0 = l_i \cdot R_i$, чи $E = 0,0,...,(l_i \cdot R_i) \cdot \bmod p_i, 0, 0, ..., 0$, оскільки тільки числа, які діляться націло на $R_i = R / p_i$ мають у своєму представленні в СЛК такий набір лишків. В останніх виразах величина $R = \prod_{i=1}^{k=n+1} p_i$ — контрольний (повний) діапазон представлення чисел у СЛК.

На числовій осі величина спотворення $l_i \cdot R_i$ відображається точкою в деякому піддіапазоні «контрольного» діапазону [(P + 1), R). Відповідно, процес спотворення початкового числа A відобразиться переміщенням точки A із робочого діапазону [0, P) до деякого піддіапазону з номером k, тобто до піддіапазону [$k \cdot P$, ($k \cdot P$ + 1)). Звернемо увагу на те, що до піддіапазону з цим номером k спотворене число ($A' = l_i \cdot R_i + A_1$ чи $A' = l_i \cdot R_i + A_2$) може попасти (див. рисунок) залежно від величини початкового числа (A_1 чи A_2) та взаємного розташування лівих границь піддіапазонів — відповідно точок $k \cdot P$ та $l_i \cdot R_i$.



Ілюстрація процесу нулізації

На рисунку (а) зображена ситуація, коли величина початкового числа A_1 перевищує різницю між значеннями $l_i \cdot R_i$ та $k \cdot P$, тобто коли $A_1 > (k \cdot P - l_i \cdot R_i)$. Ситуація, що зображена на рисунку (б) відповідає варіанту, коли величина початкового числа A_2 є меншою ніж різниця між значеннями $l_i \cdot R_i$ та $(k + 1) \cdot P$. В обох випадках результатом нулізації є число $k \cdot P$, тобто ліва границя піддіапазону із номером k, а величина спотворення $(k - 1) \cdot P < E = l_i \cdot R_i < (k + 1) \cdot P$.

Унаслідок операції нулізації із числа A', яке контролюється, віднімається число $T = A' - k \cdot P < P$. При цьому по контрольній основі p_k одержується результат γ

такий, що відповідає лівій межі (див. рисунок) піддіапазону $[k\cdot P, (k+1)\cdot P)$, тобто дорівнює величині $k\cdot P$.

Отже маємо:

$$\gamma = \{k \cdot P\}_{p_k}.$$

Звідси, за правилами СЛК, отримаємо:

$$k = \{\gamma / \{P\}_{p_k}\}_{p_k}. \qquad (1)$$

Тобто, використовуючи вираз (1), завжди можна визначити номер того діапазону, до якого потрапило спотворене число та результат нулізації — число $k\cdot P$.

Приклад 3. Визначимо номер піддіапазону k для умов Прикладу 2. Оскільки у ньому значення $\gamma = 51$, то з (1) витікає

$$\gamma = (k\cdot P) \pmod{p_к} = 51 \ (k\cdot 140) \pmod{71},$$

чи

$$k = \gamma / P(\mathrm{mod}\,71) = (\gamma + c\cdot 71)/[P \bmod 71)] = (51 + c\cdot 71)/(140 \bmod 71) = (51 + c\cdot 71)/69,$$

де величина c підбирається так, що $c = (0, 1, 2, \ldots)$ доти, поки не буде одержаним ціле значення k.

Неважко упевнитися, що при $c = 9$

$$k = (51 + 9\cdot 71)/69 = (51 + 639)/69 = 690/69 = 10,$$

і що, оскільки $\gamma = c\cdot P(\mathrm{mod}71) = 1400(\mathrm{mod}\,71) = 51$, то спотворене число при цьому попало на ліву границю діапазону $[k\cdot P, (k+1)\cdot P) = [1400,\ 1540)$.

Таким чином, для виявлення місця та величини спотворення, тобто величини основи p_i чи просто її номера i, необхідно знайти, хоча би перебором, номер тої основи i, де має місце спотворення, а величина спотворення — E задовольняє умові

$$(k-1)\cdot P < E = l_i \cdot R_i < (k+1)\cdot P.$$

Для визначення умов виявлення місця та величини спотворення розділимо всі частини цього виразу на величину P:

$$(k-1) < l_i \cdot R_i/P < (k+1), \text{ чи } (k-1) < l_i \cdot (R/p_i)/P < (k+1),$$

звідки умова виявлення місця та величини спотворення набуде вигляду

$$(k-1) < l_i \cdot (p_k / p_i) < (k+1). \qquad (2)$$

Дійсно, при переборі у виразі (2) змінних l_i (l_i = 1, 2, …, (p_i – 1)) для кожної із основ p_i (i = 1, 2, …n) можна знайти такі їхні значення, які задовольняють цим нерівностям, що є еквівалентним *виявленню місця викривлення (і) та номеру інтервалу l_i. Величину ж спотворення, знаючи величини і та l_i, неважко визначити із співвідношення:*

$$\Delta A = E = l_i \cdot R_i.$$

Приклад 4. Здійснити декодування для умов Прикладів 2 та 3 (p_1 = 4, p_2 = 5, p_3 = 7, p_k = 71), коли спотвореним числом є $\widetilde{A}$ = 11.01.01.0110011 = (3, 1, 1, 51), а значеннями констант є: $R_1 = 5 \cdot 7 \cdot 71 = 2485$, $R_2 = 4 \cdot 7 \cdot 71 = 1988$, $R_3 = 4 \cdot 5 \cdot 71 = 1420$, $R_4 = P = 4 \cdot 5 \cdot 7 = 140$. Нехай (за умовами Прикладу 2) спотворена третя пара розрядів, а після декодування (за умовами Прикладу 3) визначено $k = 10$.

Тоді для основи p_1 = 4 маємо $p_k / p_i = 17{,}75$ і, оскільки для мінімального із $l_1 = 1$ значення $p_k / p_i = 17{,}75 > 11$, то перевіряти нерівняння (2) надалі для цієї основи не має сенсу. Висновок: *спотворень у першій групі немає!*

Для основи p_2 = 5 маємо $p_k / p_i = 14{,}2$, а отже робимо ті ж висновки, що і по основі p_1: *спотворень у другій групі немає!*

Для основи p_3 = 7 маємо $p_k / p_i = 10{,}14$. Отже нерівняння (2) задовольняється при $l_3 = 1$, оскільки 1·10,14 < 11. Висновок: *виявлено спотворення в третій групі!*

Величина спотворення дорівнює $\Delta A = E = l_3 \cdot R_3 = 1 \cdot 1420 = 1420$. Звідси:

$$\Delta \alpha_3 = \Delta A (\mathrm{mod}\ p_3) = 1420 (\mathrm{mod}\ 7) = 6.$$

Здійснимо корегування виявленого спотворення

$$\alpha_3 = (\tilde{\alpha}_3 - \Delta\alpha_3) \bmod p_3 = (1 - 6) \bmod 7 = 2_{10} = 10_2,$$

що і мало місце в початковому коді ($\alpha_1 = 11_2$, $\alpha_2 = 01_2$, $\alpha_3 = 10_2$) ще на етапі кодування (див. Приклад 1).

Тобто, порівнявши отримане значення з вихідним, не спотвореним (умови Прикладу 1), упевнюємося в тому, що корекція здійснена вірно.

Надалі за скороченими процедурами надамо приклади перевірки справедливості виразу (2) для декількох спотворень у різних умовних лишках.

Приклад 5. Здійснимо декодування спотвореного числа $\widetilde{A}$ = = 10.01.10.0110011 = (2, 1, 2, 51), коли в початковому числі A = 11.01.10.0110011 спотворена перша пара розрядів.

Неважко показати, що внаслідок нулізації одержується допоміжна величина $T = \sum_{i=1}^{3} t_i$ = 86, тобто

$$T = 10.01.10.\ 1010110,$$

і результат нулізації має значення:

$$\gamma = (\alpha_k - (T \bmod p_к)) \pmod{p_к} = (0110011 - 1010110) \pmod{71} =$$
$$= (51 - 86) \pmod{71} = 36 = 0100100.$$

Оскільки $\gamma \neq 0$, то робимо *висновок про наявність спотворення в числі, що декодується*.

При визначенні номеру піддіапазону k одержимо: $k = 53$.

Перевіримо нерівняння (2) для основи $p_1 = 4$ ($p_k / p_i = 17{,}75$):

$$52 < l_i \cdot (p_k / p_i) < 54.$$

При $l_1 = 3$ маємо:

$$52 < 3 \cdot 17{,}75 = 52{,}65 < 54 \; - \underline{\text{нерівняння є вірним}};$$

Висновок: *виявлено спотворення в першій групі!*

Величина спотворення дорівнює $\Delta A = E = l_1 \cdot R_1 = 3 \cdot 2485 = 7455$. Звідси:

$$\Delta\alpha_3 = \Delta A (\bmod p_3) = 7455 (\bmod 4) = 3.$$

Корегування спотворення:

$$\alpha_1 = (\widetilde{\alpha}_1 - \Delta\alpha_1) \bmod p_1 = (2 \; - 3) \bmod 4 = 3_{10} = 11_2,$$

що і мало місце в початковому коді ($\alpha_1 = 11_2$, $\alpha_2 = 01_2$, $\alpha_3 = 10_2$) ще на етапі кодування (див. приклад 1).

Тобто, порівнявши отримане значення з вихідним, не спотвореним (умови Прикладу 1), упевнюємося в тому, що корекція здійснена вірно.

Приклад 6. Здійснимо декодування спотвореного числа $\widetilde{A}$ = = 11.11.10.0110011 = (2, 1, 2, 51), коли в початковому числі A = 11.01.10.0110011 спотворена друга пара розрядів.

Неважко показати, що в наслідок нулізації одержується допоміжна величина $T = \sum_{i=1}^{3} t_i = 86$, тобто

$$T = 10.01.10.\; 1010110,$$

і результат нулізації має значення:

$$\gamma = (\alpha_k - (T \bmod p_к)) \pmod{p_к} = (0110011 - 0010111) \pmod{71} =$$
$$= (51 - 23) \pmod{71} = 28 = 0011100.$$

Оскільки $\gamma \neq 0$, то робимо *висновок про наявність у числі, що декодується, спотворення*.

Визначимо номер піддіапазону k. Оскільки значення $\gamma = 28$, то з (2) витікає: $k = 57$.

Нерівняння (2) набуває вигляду:

$$56 < l_i \cdot (p_k / p_i) < 58 .$$

Унаслідок перевірок нерівняння (2) по основам $p_1 = 4$ ($p_k / p_i = 17{,}75$), $p_2 = 5$ ($p_k / p_i = 14{,}2$) при $l_2 = 4$ маємо:

$$56 < 4 \cdot 14{,}2 = 56{,}8 < 58 \ \text{— } \underline{\textit{нерівняння є вірним}};$$

Висновок: *виявлено спотворення в другій групі!*

Величина спотворення дорівнює $\Delta A = E = l_2 \cdot R_2 = 4 \cdot 1988 = 7952$. Звідси:

$$\Delta \alpha_3 = \Delta A (\mathrm{mod}\, p_3) = 7952 (\mathrm{mod}\ 5) = 2.$$

Корегування виявленого спотворення:

$$\alpha_2 = (\widetilde{\alpha}_2 - \Delta \alpha_2) \mathrm{mod} p_2 = (3 - 2) \mathrm{mod}\ 5 = 1_{10} = 01_2,$$

що і мало місце в початковому коді ($\alpha_1 = 11_2,\ \alpha_2 = 01_2,\ \alpha_3 = 10_2$) ще на етапі кодування (див. Приклад 1).

Висновок

На основі процедури нулізації для коду умовних лишків запропоновано алгоритм декодування, який дозволяє визначити як місце, так і величину спотворення, і за рахунок цього здійснити корекцію виявлених спотворень.

1. *Матов О.Я.* Узагальнені завадостійкі коди в задачах забезпечення цілісності інформаційних об'єктів. Код умовних лишків / О.Я. Матов, В.С. Василенко // Реєстрація, зберігання і оброб. даних. — 2006. — Т. 8, № 3. — С. 48–66.

2. *Василенко В.С.* Вибір величини контрольної основи для коду умовних лишків / В.С. Василенко, О.Я. Матов // Реєстрація, зберігання і оброб. даних. — 2010. — Т. 12, № 1. — С. 73–78.

3. *Василенко В.С.* Алгоритми кодування-декодування узагальнених завадостійких кодів для забезпечення цілісності в умовах природних впливів / В.С. Василенко // Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні. — 2008. — Вип. 2(17). — С. 56–66.

Надійшла до редакції 30.06.2011