

The basic notions in relation to the marketing researches are considered. The systematization of the types of the marketing researches, covering the different factors and conditions, directly affecting the processes of organization of the entrepreneurial activity is represented. It is proved that the usage of the marketing information system, as the basic tool for making marketing decisions allows applying the aggregate choice of the different systems, methods and tools, at assistance of which the internal and external information is analysed and interpreted.

**Key words:** marketing environment, market, sales, economy, information, enterprise, motivation, staff, processing, target, marketing information system.

**УДК 681.3** *Доц. Л.В. Мороз, д-р техн. наук; бакалавр Я.І. Грабовський; аспір. Т.М. Микитів; аспір. Т.Р. Борецький; асист. Ю.М. Костів, канд. техн. наук; ст. викл. С.С. Войтусік, канд. фіз.-мат. наук – НУ "Львівська політехніка"*

### ШВИДКОДІЮЧИЙ ГІБРИДНИЙ CORDIC-ОБЧИСЛЮВАЧ ТРИГОНОМЕТРИЧНИХ ФУНКЦІЙ

Розглянуто спосіб обчислення тригонометричних функцій, запропонований В.Д. Байковим, як одна з перших спроб застосувати пам'ять для підвищення швидкодії традиційного CORDIC алгоритму. Недоліком цього методу є значна похибка результату на певних критичних ділянках. Запропоновано декілька нових підходів, що забезпечують підвищення швидкодії за умови збереження точності обчислень. Одним із них є використання кінцевих помножувачів після проведення половини ітерацій. Розроблено гібридний метод, що поєднав у собі три складові: використання пам'яті – класичний CORDIC – кінцеві помножувачі. При цьому, за результатами моделювання на мікроконтролері, досягнуто підвищення швидкодії у 2,9 раза, порівняно з традиційним CORDIC алгоритмом без втрати точності.

**Ключові слова:** синус, косинус, CORDIC, помножувач, гібридний метод.

**Вступ.** CORDIC-метод (або метод псевдоповороту вектора, метод Волдера) – це простий та ефективний метод обчислення багатьох елементарних функцій. Назва методу походить від англійського COordinate Rotation Digital Computer – цифровий комп'ютер повороту координат. Метод популярний і нині завдяки простоті його програмної та апаратної реалізації. Причиною популярності є те, що для класичного CORDIC-методу потрібен лише невеликий обсяг пам'яті та деякі елементарні операції, типу читання з пам'яті, додавання / віднімання та зсув. CORDIC-метод використовується в багатьох галузях, таких як: 3D графіка, робототехніка, модулі зв'язку, системи біометричної ідентифікації та ін. Він є надзвичайно перспективним, оскільки може бути застосований практично в кожній галузі, у якій використовуються лінійні, матричні й тригонометричні обчислення. Теорія та алгоритми методу розглянуто в значній кількості книгах та працях багатьох авторів [3, 5, 6, 8, 9, 13].

Головною концепцією CORDIC-арифметики є те, що її засновано на простих принципах двовимірної геометрії. Ітераційні формулювання алгоритмів методу були вперше описані в 1959 р. Джеком Волдером (Jack E. Volder) [11] під час обчислення тригонометричних функцій, реалізації операцій множення і ділення. За весь час існування CORDICа відбувся великий прогрес у напрямі розроблення алгоритмів і архітектур для високопродуктивних та дешевих апа-

ратних рішень. Увага до обчислень, що базуються на CORDIC-методі, підвищилась у 1971 р., коли Джон Велзер (John Walther) [12] продемонстрував, що шляхом зміни декількох параметрів можна обчислювати елементарні трансцендентні функції, логарифми, експоненційну функцію та квадратний корінь [11].

Популярність CORDICа пов'язана насамперед з його потенціалом до ефективного та дешевого впровадження у такі задачі, які включають: генерацію тригонометричних, логарифмічних та інших елементарних функцій; множення комплексних чисел, обчислення власних значень та сингулярне розкладання (SVD) матриць, розв'язування систем лінійних рівнянь тощо.

Хоча CORDIC-метод можливо і не найшвидший технічний прийом для виконання цих операцій (метод має лише лінійну збіжність), але він є простим у впровадженні на фізичному рівні, оскільки один і той же алгоритм можна використати для всіх цих додатків, застосовуючи базові операції зсуву та додавання [4, 10].

Незважаючи на різноманітність різних технік, запропонованих для прискорення збіжності алгоритмів методу, майже всі вони зберігають його ітераційний характер. Прискорення досягається переважно завдяки зменшенню загальної кількості ітерацій. Відома спроба підвищити швидкодію класичного CORDIC-методу завдяки використанню ПЗУ на ранніх стадіях обертань, однак вона не мала практичного застосування внаслідок виникнення при цьому значної похибки обчислень [1]. Аналогічно ще в 1972 р. [2] було запропоновано спосіб використання операції множення на залишковий кут на кінцевих стадіях обертань, який повторено за кордоном лише у 1991 р. і пізніше [7, 9]. Однак і ця ідея була практично втілена у ПЛІС-варіанті лише недавно [14].

**Метод Байкова.** Спосіб, який передбачає використання табличного методу (ПЗП-LUT) на початкових стадіях ітераційного процесу, вперше був описаний В.Д. Байковим у роботі [1]. Тут  $m$ -розрядний аргумент  $\phi_m$  (вхідний кут, заданий в радіанах) поділяється на дві частини:

$$\phi_m = \phi_s + \phi_{m-s}, \quad (1)$$

де:  $\phi_s$  – старша частина, що містить  $s$  двійкових розрядів;  $\phi_{m-s}$  – молодша частина, що містить решту  $m-s$  розрядів. Для спрощення викладу матеріалу можна вважати, що  $m$  – парне. Передбачається, що  $\phi_m \in [0, \pi/4]$ , тому

$$\phi_s = \sum_{i=1}^s a_i 2^{-i}, \quad (2)$$

$$s < m/2.$$

Старші розряди обробляються за допомогою ПЗП, а решта – за допомогою класичного CORDIC-методу:

$$\begin{aligned} x_{i+1} &= x_i - \sigma_i \cdot 2^{-i} \cdot y_i; \\ y_{i+1} &= y_i + \sigma_i \cdot 2^{-i} \cdot x_i; \\ z_{i+1} &= z_i - \sigma_i \cdot \theta_i; \end{aligned} \quad (3)$$

де  $\sigma_i = \begin{cases} -1 & \text{if } z_i < 0 \\ +1 & \text{if } z_i \geq 0 \end{cases}, i = s+1, s+2, \dots, m, \theta_i = \arctg(2^{-i})$ .

При цьому значення, що завантажуються з пам'яті, обчислюються за формулами:

$$\begin{aligned} x_s &= P \cdot \cos(\phi_s); \\ y_s &= P \cdot \sin(\phi_s); \\ z_s &= \phi_{m-s}; \end{aligned} \quad (4)$$

де  $P$  – величина, обернена до коефіцієнта деформації:

$$P = 1 / \prod_{i=s+1}^m \sqrt{1 + 2^{-2i}}. \quad (5)$$

Отож, тут початкові значення  $x_s$  та  $y_s$  уже є скориговані на значення коефіцієнта деформації вектора з урахуванням необхідних ітерацій традиційного CORDIC-методу.

Цей метод має істотні недоліки, оскільки не забезпечує необхідної точності на всьому діапазоні вхідних значень, що буде продемонстровано у результатах моделювання.

**Метод із використанням кінцевих помножувачів.** Одним із методів прискорення швидкодії є використання кусково-лінійної апроксимації функцій синуса та косинуса на завершальних стадіях ітераційного процесу шляхом заміни останніх  $k = m / 2$  ітерацій множенням на залишковий кут.

У цьому випадку на кроках  $i = 1 \dots k$  використовуються традиційні формули (3) з вхідними даними:

$$\begin{cases} x_0 = P_k \\ y_0 = 0 \\ z_0 = \phi_m \end{cases} \quad (7)$$

коефіцієнт корекції  $P_k$  обчислюється тільки для тих значень  $i$ , що задовольняють умову (6):

$$P = 1 / \prod_{i=1}^k \sqrt{1 + 2^{-2i}}. \quad (8)$$

Після цього виконуються дві операції:

$$\begin{aligned} x_{k+1} &= x_k - z_k \cdot y_k, \\ y_{k+1} &= y_k + z_k \cdot x_k, \end{aligned} \quad (9)$$

де  $x_k$ ,  $y_k$ ,  $z_k$  – значення, які отримані на попередньому етапі роботи.

Застосування кінцевих помножувачів дає змогу істотно (приблизно у два рази) зменшити число тактів процесора (табл.).

**Табл. Результати моделювання**

Алгоритм	Максимальна додатна похибка		Максимальна від'ємна похибка		Кількість тактів
	sin	cos	sin	cos	
Традиційний CORDIC	$7,1 \cdot 10^{-9}$	$5,1 \cdot 10^{-9}$	$-6,8 \cdot 10^{-9}$	$-5,7 \cdot 10^{-9}$	792
Метод Байкова ПЗП – 4 біти	$1,1 \cdot 10^{-5}$	$7,7 \cdot 10^{-6}$	$-1,0 \cdot 10^{-5}$	$-5,7 \cdot 10^{-9}$	706
Залишкового множення	$3,7 \cdot 10^{-9}$	$3,2 \cdot 10^{-9}$	$-4,2 \cdot 10^{-9}$	$-4,4 \cdot 10^{-9}$	462
Гібридний, ПЗП – 4 біти	$4,1 \cdot 10^{-9}$	$4,1 \cdot 10^{-9}$	$-3,3 \cdot 10^{-9}$	$-3,2 \cdot 10^{-9}$	369
Гібридний, ПЗП – 8 біт	$4,1 \cdot 10^{-9}$	$3,7 \cdot 10^{-9}$	$-3,1 \cdot 10^{-9}$	$-3,1 \cdot 10^{-9}$	273

**Гібридний метод.** Запропоновано використовувати всі описані вище методи в такій послідовності: табличний – класичний CORDIC – вихідні помножувачі. Спочатку знаходяться значення  $x_s$ ,  $y_s$  і  $z_s$  за формулами (4), потім проводяться ітерації за формулами класичного CORDIC-методу:

$$\begin{aligned} x_{i+1} &= x_i - \sigma_i \cdot 2^{-i} \cdot y_i; \\ y_{i+1} &= y_i + \sigma_i \cdot 2^{-i} \cdot x_i; \\ z_{i+1} &= z_i - \sigma_i \cdot \theta; \end{aligned} \quad (10)$$

де  $\sigma_i = \begin{cases} -1 & \text{if } z_i < 0 \\ +1 & \text{if } z_i \geq 0 \end{cases}$ ,  $i = s+1, s+2, \dots, m/2$ ,  $\theta_i = \arctg(2^{-i})$ , а після завершення цих ітерацій використовуються вихідні множення на залишковий кут відповідно до формули (9). Таке поєднання дає змогу істотно підвищити швидкодію без втрати точності обчислень, оскільки застосування способу вихідних помножувачів допомагає врахувати відсутній кут під час використання табличного методу.

**Отримані результати.** Тестування проведено на мікроконтролері STM32F100RBT6B. Це мікроконтролер на ядрі ARM Cortex-M3, що розробляється для додатків, де важливі висока продуктивність, низька ціна й енергоспоживання, час реакції на переривання, а також простота у використанні.

У мікроконтролері є 16 регістрів. Один регістр використовується як покажчик стека, інший – як лічильник команд, ще один зберігає адресу повернення після виклику підпрограми. Інші 13 регістрів призначені для загального користування. Усі регістри є 32-бітними. Також мікроконтролер має 32-бітну шину даних і 32-бітові інтерфейси пам'яті.

Мікроконтролер виконано за гарвардською архітектурою, у якій розділені шини команд і даних. У самому ядрі використовується фіксований розподіл адресного простору. Є область коду програми, статичного ОЗП, пристроїв введення – виведення і системних ресурсів.

Для перевірки теоретичних результатів змодельовано методи обчислень:

- а) традиційний CORDIC (3);
- б) Байкова (3)-(5) – з використанням ПЗП-LUT на початку роботи;
- в) залишкового множення (9);
- г) гібридний (запропонований) із розміром переглядової таблиці  $N = 2^4$ ;
- д) гібридний із розміром таблиці  $N = 2^8$  значень.

Критерієм швидкодії алгоритму виступає оцінка усередненого значення кількості тактів мікроконтролера для виконання підпрограми обчислення пари функцій  $\sin(x)$  та  $\cos(x)$  на інтервалі  $[-\pi/4; \pi/4]$ . Критерієм правильності роботи методу є оцінювання величини відхилення отриманих значень тригонометричних функцій від еталонного значення, обчисленого з вищою точністю за допомогою пакета Matlab 7.1. У таблиці наведено абсолютну додатну та від'ємну похибки обчислення для кожної функції, а також середню кількість тактів роботи підпрограми обчислення.

**Висновки.** З розглянутих методів традиційний CORDIC має найнижчу швидкодію. Метод Байкова за умови вищої швидкодії (рис. 1-2) містить критичні ділянки, на яких значно (приблизно у два рази) знижується, точність тому його можна використати на практиці. Метод залишкового множення забезпечує

необхідну точність обчислень на всьому діапазоні за умови зростання швидкодії в 1,71 раза, порівняно з традиційним CORDIC.

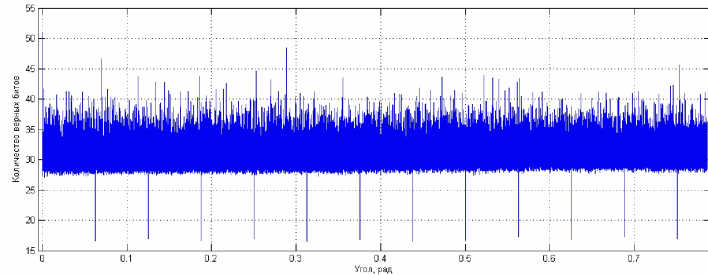


Рис 1. Метод Байкова. Кількість вірних біт, функція синус

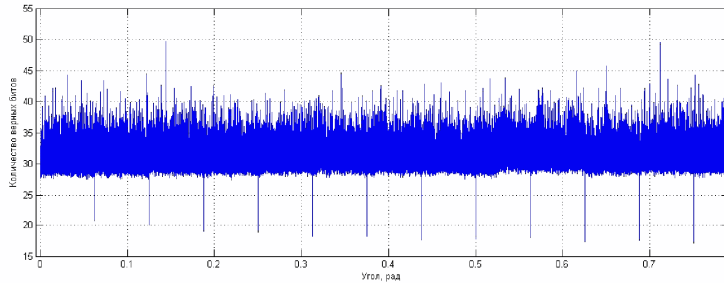


Рис 2. Метод Байкова. Кількість вірних біт, функція косинус

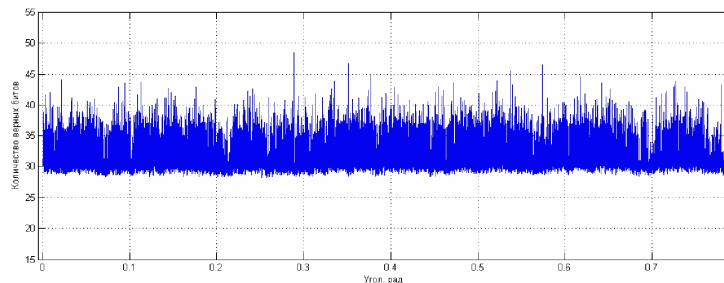


Рис 3. Гібридний метод. Функція синус

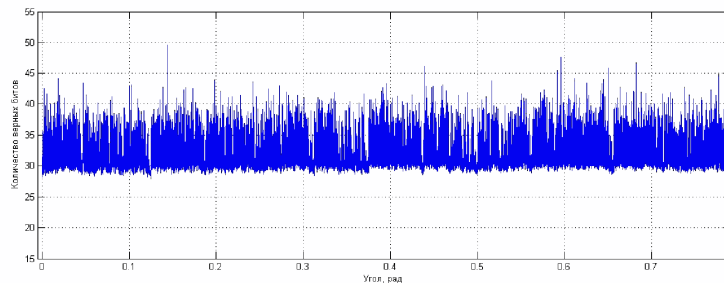


Рис 4. Гібридний метод. Функція косинус

Застосування запропонованого гібридного методу дає змогу підвищити швидкодію у 1,69 раза, порівняно з методом залишкового множення (рис. 3-4). Гібридний метод можна застосувати в мікроконтролерних системах, де необхідна висока швидкодія. Метод залишкового множення краще підходить для обчислювальних систем, побудованих на ПЛІС, оскільки потребує мінімальної кількості пам'яті за умови достатньо високої швидкодії.

### Література

1. Байков В.Д. Специализированные процессоры: итерационные алгоритмы и структуры / В.Д. Байков, В.Б. Смолов. – М.: Изд-во "Радио и связь", 1985. – 288 с.
2. Благовещенский Ю.В. Вычисление элементарных функций на ЭВМ / Ю.В. Благовещенский, Г.С. Теслер. – К.: Изд-во "Техника", 1977. – 208 с.
3. Arndt J. Matters computational / J. Arndt // Ideas, algorithms, source code, 2010. – Pp. 966.
4. Chi-Chia, S. VLSI Design Concepts for Iterative Algorithms / S. Chi-Chia // PhD thesis. Technischen Universit'at Dortmund. 11.04.2011.
5. Ercegovac M.D. Digital Arithmetic Kluwer / M.D. Ercegovac, T. Lang. – Academic Publishers, 2004.
6. Koren I. Computer arithmetic algorithms / I. Koren. Second edition. 2002. – 266 p.
7. Meher P.K. 50 years of CORDIC: algorithms, architectures, and applications / P.K. Meher, J. Valls, T. Juang, K. Sridharan, K. Maharatna. – IEEE Transactions on Circuits and Systems – I: Regular Papers. – Vol. 56, № 9, September, 2009. – Pp. 1893-1907.
8. Muller J.-M. Elementary functions / J.-M. Muller // Algorithms and implementation. Second edition – USA : Birkhäuser, 2006. – Pp. 265-269.
9. Parhami B. Computer arithmetic / B. Parhami // Algorithms and hardware designs, 2000. – 234 p.
10. Paska B. High-performance floating-point computing on reconfigurable circuits – PhD thesis. Septembre 2011. – 286 p.
11. Volder J.E. The Cordic trigonometric computing technique. – IBE Trans, on Electronic Comp, 1959. – Vol. 8, № 3. – Pp. 330-334.
12. Walther J.S. A unified algorithm for elementary functions. – Proc. of Spring Joint Corp. Conf. (SJCC), New Jersey, 1971. – Vol. 38. – Pp. 379-385.
13. Zarowski C.J. An introduction to numerical analysis for electrical and computer engineers. – C.: Wiley-Interscience, 2004. – Pp. 586.
14. F. de Dinechin, M. Istoan, and G. Sergent, "Fixed-point trigonometric functions on FPGAs", in *Proceedings of the 4<sup>th</sup> International Symposium on Highly-Efficient Accelerators and Reconfigurable Technologies*, Royaume-Uni, Edimburgh, UK, March 2013.

### Мороз Л.В., Грабовский Я.И., Мыкитив Т.М., Борецкий Т.Р., Костив Ю.М., Войтусик С.С. Быстродействующий гибридный CORDIC-вычислитель тригонометрических функций

Рассмотрен метод Байкова для вычисления тригонометрических функций, как одна из первых попыток применить память для повышения быстродействия традиционного CORDIC алгоритма. Недостатком метода является значительная погрешность результата на критических участках. Предложены несколько новых подходов, обеспечивающих повышение быстродействия при сохранении точности вычислений. Одним из них является использование конечных умножителей после проведения половины итераций. Разработан гибридный метод, который соединил три составляющие: использование памяти – классический CORDIC – конечные умножители. При этом, по результатам моделирования на микроконтроллере, достигнуто повышение быстродействия в 2,9 раза по сравнению с традиционным CORDIC алгоритмом без потери точности.

**Ключевые слова:** синус, косинус, CORDIC, умножитель, гибридный метод.

### Moroz L.V., Grabowsky Ya.I., Mykytiv T.M., Boretsky T.R., Kostiv Yu.M., Voytusik S.S. High-Speed Hybrid CORDIC-Calculator for Trigonometric Functions

The paper presents Baykov method for calculating trigonometric functions as one of the first attempts to use the memory in order to improve the performance of traditional CORDIC algorithm. The disadvantage of this method is supposed to be the result of a significant accuracy in the critical areas. The authors have proposed several new approaches for improving performance while retaining accuracy of the calculations. One of them is the use of final multipliers after half iterations. Also, it was possible to develop a hybrid method, which connects three components: memory usage – classic CORDIC – final multipliers. In this case, the results of simulation on the microcontroller, achieved speedup factor of 2.9 compared to the conventional CORDIC algorithm without loss of accuracy.

**Key words:** sine, cosine, CORDIC, multiplier, hybrid method.

УДК 630.32.002.5(075.8)

Асист. Ю.І. Цимбалюк;

проф. М.Г. Адамовський, канд. техн. наук – НЛТУ України, м. Львів

### ІМІТАЦІЙНА МОДЕЛЬ РУХУ МАЛОГАБАРИТНОЇ ТРЕЛЮВАЛЬНОЇ СИСТЕМИ

Розроблено розрахункову модель штучно створеного лісового насадження з урахуванням основних чинників, які впливатимуть на параметри малогабаритної трелювальної системи. Розглянуто трелювання лісоматеріалів під наметом лісу та розроблено імітаційну модель руху малогабаритної трелювальної системи на базі причіпних засобів без урахування тягової одиниці. Виконано моделювання руху трелювальної системи для трелювання лісоматеріалів у напівзавантаженому та напівзбалансованому стані. Показано порядок роботи та можливості створеної імітаційної моделі для практичного і наукового застосування.

**Ключові слова:** круглий лісоматеріал, трелювання, імітаційна модель, трелювальна система, рубки догляду.

**Актуальність роботи.** З розвитком комп'ютерних технологій з'явилась можливість більш широкого застосування імітаційного моделювання в лісовій галузі, зокрема для моделювання окремих операцій лісозаготівельного процесу. Однією з таких операцій є трелювання деревної сировини під час виконання доглядових рубань без прокладання розгалуженої мережі транспортних шляхів. Від цієї операції значною мірою залежить загальний рівень негативного техногенного впливу на лісове середовище. Виходячи з цього, актуальним є дослідження впливу параметрів лісового насадження та окремих технологічних чинників на параметри малогабаритної трелювальної системи для її безперешкодного руху в лісовому насадженні з урахуванням мінімального пошкодження ростучих дерев.

Проведення таких досліджень в реальних умовах ускладнюється тим, що для кожної серії експериментів, згідно зі складеним планом, необхідно шукати лісові насадження з відповідними параметрами й умовами місцезростання, щоб забезпечити умови експерименту. Тому в таких випадках актуальним стає часте використання імітаційного моделювання, яке на сучасному етапі розвитку комп'ютерних технологій забезпечує широкі можливості проведення досліджень з мінімальними матеріальними затратами і в короткі терміни.

**Аналіз останніх досліджень і публікацій.** Аналіз публікацій з цієї тематики [1, 2] показує, що проводяться роботи, пов'язані з імітаційним моделюванням руху лісової машини під наметом лісу з метою дослідження проникності

або доступності лісового насадження для цих машин. Отримані висновки показують, що розбіжності між імітаційними дослідженнями та дослідженнями в реальних умовах не перевищують 10 %.

Однак існуючі імітаційні моделі не враховують особливості малогабаритних трелювальних систем на базі причіпних засобів і розроблені для доглядових рубань в насадженнях природного походження. Комп'ютерне моделювання широко впроваджується для планування різних видів рубок, зокрема доглядових рубань [3, 4]. Це дає змогу розв'язувати великий спектр технологічних задач, пов'язаних із процесом лісозаготівлі та ставить планування доглядових рубань на новий якісний рівень високих технологій.

**Мета роботи** – розроблення імітаційної моделі руху малогабаритної трелювальної системи на базі причіпних засобів під наметом штучно створеного лісового насадження, яку можна використовувати для проведення експериментальних досліджень.

**Основний матеріал.** Зважаючи на те що, половина лісів України є штучно створені [5] й намітилася тенденція до подальшого збільшення їх площі, потрібно розглянути рух малогабаритної трелювальної системи у штучно створеному рядному лісовому насадженні з відомою схемою розміщення дерев і таксаційними показниками насадження. Такий рух відбувається під час виконання доглядових рубань без прокладання розгалуженої мережі трелювальних волоків і технологічних коридорів. Для досягнення поставленої мети було складено алгоритм програми, за яким розроблено комп'ютерну імітаційну модель.

Побудова імітаційної моделі зводилась до розв'язку таких основних задач:

- генерування моделі штучно створеного лісового насадження відповідно до зазначених вхідних параметрів у прямокутній системі координат;
- математичний опис траєкторій руху початкової "ведучої" і кінцевої "веденої" точок трелювальної системи залежно від способу трелювання круглого лісоматеріалу та заданої довжини трелювальної системи, а також перевірка можливості руху в цьому насадженні.

Для генерування штучно створеного лісового насадження за основу прийнято його розрахункову модель (рис. 1). Основними параметрами насадження прийнято  $b$  – ширину міжряддя і  $a$  – відстань між деревами в рядах.

При цьому враховано можливе відхилення окремих дерев або рядів від запланованої схеми посадки, ці відхилення враховуються показниками:  $\Delta b$  – можливе зміщення рядів;  $\Delta b_1$  – можливе зміщення окремого дерева в сторону міжряддя (випадання дерева з ряду);  $\Delta a$  – зміщення між деревами в сусідніх рядах;  $\Delta a_1$  – можливе зміщення окремого дерева в ряду. У моделі враховується також середній діаметр  $d_{cp}$  дерев у місці зрізання. Кут  $\beta$  визначає початкове положення трелювальної системи перед початком руху, цей кут прийнято за кут напрямку звалювання дерева. За ширину трелювальної системи прийнято ширину трелювального засобу  $b_0$ .

Для генерування лісового насадження за допомогою створеної програми і візуалізації результатів на екрані комп'ютера вводяться такі дані:

- діаметр дерев у місці зрізування;
- відстані між деревами в ряду і між рядами;
- граничні межі відхилення від схеми посадки дерев у ряду і між рядами.