

The heuristic function for comparing polar angles of the vectors defined in two-dimensional rectangular coordinate system

Volodymyr Semeniakin, Tetiana Zabolotnia
Computer Systems Software Department, National Technical
University of Ukraine "Kyiv Polytechnic Institute",
UKRAINE, Kyiv, Peremogy avenue, 37,
E-mail: tatiana104@yandex.ru, semenyakins@rambler.ru

The development of the software connected with vectors processing often requires vectors data recalculation from the rectangular coordinate system into the polar one. As an example of the task that uses such recalculation we can specify the task of fields of view searching in the two-dimensional space or the problem of ray tracing for the three-dimensional case.

At the same time it's necessary to notice that recalculation algorithms require the execution of resource-intensive actions such as the square root calculation and the antitrigonometric function computation. Using of these functions is unallowable in case of performing tasks that need minimal CPU load.

This paper is devoted to the question of the reducing of run time of the algorithms of the rectangular coordinates recalculation into the polar coordinates by using equivalent functions that are functions that behave similar to the common recalculation functions. The values returned by equivalent functions may not coincide with the values of the common functions results but despite it their using allows to conduct comparative analysis of the appropriate coordinates.

In the given work the question of using equivalent function for vector length index computation is considered briefly, the main part of the paper is devoted to the search of the equivalent function for getting the angle value.

The sound requirements to the developed equivalent function are presented. On that basis the tangent function was selected to be the basis of the new equivalent function.

The main steps for bring the group of tangent functions to the appearance, fully satisfying the brought forward requirements, are proposed.

At the end of paper the run time indicators for the recalculation of coordinates using common function of the STL library and the ACV function, proposed in the thesis, are represented.

Евристична функція порівняння полярних кутів векторів, заданих у двовимірній декартовій системі координат

Володимир Семенякін, Тетяна Заболотня

Кафедра програмного забезпечення комп'ютерних систем,
Національний технічний університет України "Київський
політехнічний інститут",
УКРАЇНА, м.Київ, пр.Перемоги, 37,
E-mail: tatiana104@yandex.ru, semenyakins@rambler.ru

У доповіді розглядається питання пошуку евристичної функції для зменшення обчислювальної складності алгоритмів, що потребують виконання порівняння значень полярних кутів векторів, координат яких подані у декартовій системі координат. Основною ідеєю функції є зменшення складності математичних дій, що необхідні для виконання порівняння значень полярних кутів векторів за умови збереження коректності результату цього порівняння. У доповіді запропоновані вимоги до шуканої функції, на основі яких викладено послідовність міркувань щодо способу визначення евристичної функції. Для доведення переваг нової функції у доповіді приводяться результати виконання розробленої функції і функції STL для обчислення значення полярного кута вектору за його двовимірними декартовими координатами.

Ключові слова – евристика, високопродуктивні обчислення, полярні координати.

I. Вступ

Під час створення програмного забезпечення обробки графіки та геометричних даних в цілому часто виникає необхідність порівняння полярних координат. Наприклад:

- двовірний випадок – програма визначення областей видимості [1], для j -сортування (алгоритму порівняння точок за їх кутовою координатою у полярній системі координат);
- тривірний випадок – трасувальник променів, де також може виникати необхідність у порівнянні значень полярних кутів.

Для отримання значення полярних координат точки з декартових двовірних координат використовуються: операція нормування вектора і функція арктангенс [2]:

$$r = \sqrt{x^2 + y^2} \quad (1)$$

$$j = \begin{cases} \arctg(y/x), & x > 0, y \geq 0 \\ \arctg(y/x) + 2p, & x > 0, y < 0 \\ \arctg(y/x) + p, & x < 0 \\ p/2, & x = 0, y > 0 \\ 3p/2, & x = 0, y < 0 \end{cases} \quad (2)$$

Операції взяття кореня і арктангенса є досить складними для розрахунку на ЕОМ. Взяття кореня вимагає кількох ділень [3], функція арктангенс –

розкладання в ряд і виконання ітераційних обчислень [4], що потребує більшого часу для виконання, ніж елементарна арифметична операція (додавання, множення, ділення).

Звернімо увагу, що в задачах, де потрібно виконувати тільки операції порівняння координат точок за показниками $\underline{j}$ та $\underline{r}$, немає необхідності знати точні значення цих показників. У таких випадках замість класичних функцій обчислення параметрів $\underline{j}$ та $\underline{r}$ доцільно використовувати еквівалентні їм за поведінкою евристичні функції (далі – еквівалентні функції, позначення: $\mathbb{J}$ та $\mathbb{R}$), що потребують меншої кількості операцій, а отже, працюють більш ефективно у сенсі часу виконання на ЕОМ.

Очевидно, при обчисленні параметру $\mathbb{R}$ можливо уникнути обчислення квадратного кореня, оскільки $\sqrt{x_1^2 + y_1^2} > \sqrt{x_2^2 + y_2^2}$, $x_1^2 + y_1^2 > x_2^2 + y_2^2$, тому що $x_1^2 + y_1^2 > 0$, $x_2^2 + y_2^2 > 0$, де $x_1, y_1, x_2, y_2 \in \mathbb{R}$.

Ідея використання подібної еквівалентної функції не є новою [5]. Питання ж отримання еквівалентної функції для ефективного за часом виконання розрахунку параметра $\mathbb{J}$ є недослідженим і тому актуальним. З огляду на це дана доповідь присвячена розв'язанню задачі визначення ефективного за часом виконання на ЕОМ способу обчислення еквівалента $\mathbb{J}$.

II. Евристична функція порівняння полярних кутів векторів

Встановимо, які властивості повинна мати шукана еквівалентна функція від двох змінних (координат x та y вектора, для якого здійснюється обчислення полярних координат). При цьому будемо використовувати формули отримання декартових координат векторів з полярних:

$$x = r \cos(j) \quad (3)$$

$$y = r \sin(j) \quad (4)$$

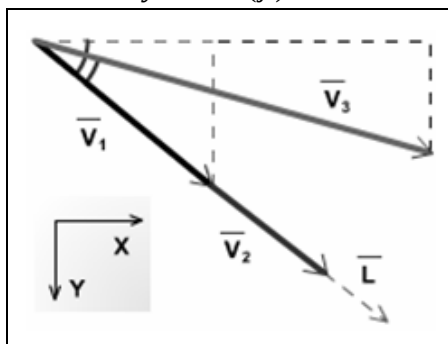


Рис.1 Ілюстрація до переліку властивостей функції $\mathbb{J}$

1. Значення, що повертаються шуканою функцією для векторів, які лежать на одному промені, повинні бути однаковими (на Рис. 1 $\underline{\bar{V}}_1$ та $\underline{\bar{V}}_2$ лежать на одному промені $\underline{\bar{L}}$).

2. Шукана функція $\mathbb{J} = F(x, y)$: виходячи з формул (3) та (4), можна вважати, що $\mathbb{J} = F(r \cos(j), r \sin(j)) = F_1(j)$, як і лінійна функція $F(j) = j$, має бути монотонно зростаючою та неперервною на проміжку $j \in [0; 2p)$ (на Рис. 1 $\mathbb{J}$ для $\underline{\bar{V}}_3$ менше, ніж для $\underline{\bar{V}}_1$ та $\underline{\bar{V}}_2$, так само, як і параметр j для цих векторів).

Встановимо найбільш ефективну з точки зору часу виконання функцію, для якої виконується перша властивість шуканої функції. Такою функцією є тангенс кута j , який можна отримати шляхом ділення координат вектора. Це легко довести, якщо скористатись формулами (3) та (4):

$$y/x = \frac{r \sin(j)}{r \cos(j)} = \frac{\sin(j)}{\cos(j)} = \tan(j) \quad (5)$$

Жодна інша елементарна операція (додавання, віднімання, множення) не дає результату, відповідного першій властивості шуканої функції. Використання ж декількох операцій спричинить втрату швидкості виконання розрахунку функції, що є неприпустимим. Отже, $F(x, y) = y/x$ характеризується першою властивістю, але не задовольняє другій, тому що при $x = 0$ має місце розрив.

Щоб уникнути цієї проблеми, було вирішено використовувати функцію, аналогічну за поведінкою $F(x, y) = y/x$ – функцію котангенс, взяту зі знаком мінус (знак мінус необхідно брати задля виконання умови монотонного зростання функції, оскільки функція котангенс монотонно зменшується на проміжках безперервності), $F(x, y) = -x/y$. На відміну від тангенса, мінус-котангенс має розрив при $y = 0$.

Тепер, якщо задати $\mathbb{J}$ кусково, визначаючи як тангенс або котангенс від j поперемінно на проміжках, де дві зазначені функції не мають розривів, можна домогтися виконання другої властивості шуканої функції – властивості монотонного безперервного зростання на проміжку $j \in [0; 2p)$.

При такому підході слід обережно обирати проміжки, на яких буде відбуватися зміна функцій – розриви і стрибки значень можуть виникати внаслідок кускової природи $\mathbb{J}$. Щоб уникнути цього, слід відштовхуватися від факту, що при $x = y$ та $j \in [0; 2p)$, $y/x = x/y = 1$, а, отже, умовою зміни функції доцільно взяти умову $x = y$.

При цьому, для задоволення вимоги відсутності розривів, необхідно замість $F(x, y) = -x/y$ брати функцію $F(x, y) = 2 - x/y$, тому що $x/y \in [0, 1]$ при $j \in [p/4; p/2)$ і, отже, функція $\mathbb{J} = 2 - x/y \in [1, 2]$ при $j \in [p/4; p/2)$. Зауважимо, що $y/x \in [0, 1]$ при $j \in [0; p/4)$. Таким чином, на проміжку $j \in [0; p/2)$ шукана функція $\mathbb{J}$ буде задана як:

$$j = \begin{cases} y/x, & \text{при } x < y, x > -y, y > 0 \\ 2 - x/y, & \text{при } x > y, x > -y \end{cases} \quad (6)$$

Подібний підхід можна застосувати до всього проміжку визначення функції, тобто $j \in [0; p/2)$.

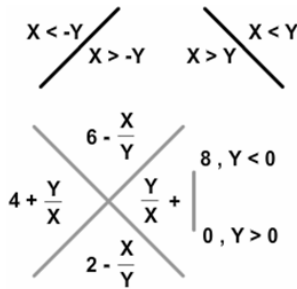


Рис.2 Ілюстрація значень функції j

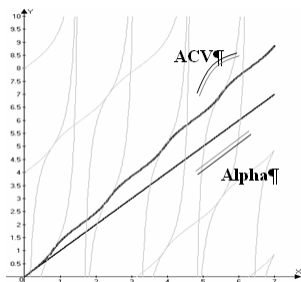


Рис.3 Графіки функцій ACV та j

На рис. 2 показаний результат пошуку функції: які значення повинна приймати шукана функція в залежності від значення поточних координат x та y . На рис. 3 подано отриману функцію і лінійну функцію $F(j) = j$ (світло-сірими лініями зображені повні графіки кусково заданих функцій, якими визначена функція j).

Примітка: як аргументи функції j на графіку взяті значення $x = r \cos(j)$, $y = r \sin(j)$, $j \in [0, \pi/2]$, r - деяка константа (не впливає на вигляд графіка внаслідок першої властивості функції j).

На рисунку видно, що знайдена функція є не тільки безперервною і монотонно зростаючою, а й гладкою, бо дотичні для функцій на сусідніх внутрішніх проміжках рівні. Назву функції було вирішено дати за скороченням слів «значення для порівняння кутів» **Angle Comparing Value – ACV**.

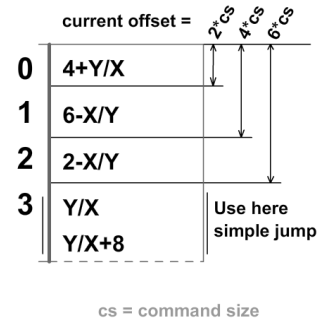
Алгебраїчний запис:

$$ACV(x, y) = \begin{cases} y/x, & x < y, x > -y, y \geq 0 \\ 2 - x/y, & x > y, x > -y \\ 4 + y/x, & x > y, x < -y \\ 6 - x/y, & x < y, x < -y \\ 8 + y/x, & x < y, x > -y, y < 0 \end{cases} \quad (7)$$

При реалізації функції ACV на Асемблері можливо запобігти зайвих розгалужень у програмі за допомогою використання спеціальної змінної, що зберігатиме зміщення інструкцій асемблера конкретної формули обчислення функцій.

$X > -Y$		$X < Y$	
0 / 1	0 / 1		
0	0	$X > Y$ $X < -Y$	0
0	1	$X < Y$ $X < -Y$	1
1	0	$X > Y$ $X > -Y$	2
1	1	$X < Y$ $X > -Y$	3

Рис.4 Залежність значення зміщення (offset) від значень X та Y



cs = command size

Рис.5 Ілюстрація розміщення інструкцій на обчислення у пам'яті

Для цього з початку необхідно встановити максимальну кількість дій, що їх потрібно виконати для отримання значення функції на кожному з проміжків значення аргументів (на рис. 5 цей параметр позначено як command size). Після цього, за виконання умови $X < Y$, треба додати цю константу до значення поточного зміщення (на рис. 5 – current offset) і виконати зміщення вказаної константи на один разряд ліворуч.

Після цього, за виконання умови $X > -Y$, треба додати зміщенню константу до значення поточного зміщення. Отримавши таким чином повне значення поточного зміщення, можна виконати стрибок по цьому зміщенню.

Запропонована вище реалізація функції сприяє зменшенню кількості розгалужень у програмі і, внаслідок цього, зменшує кількість кеш-промахів під час її обчислення.

Висновок

Таким чином, в результаті проведення описаних у доповіді досліджень була отримана функція, що дозволяє за дві операції порівняння (за три при $X < Y$, $X > -Y$), операцію суми та операцію ділення отримати значення, спираючись на яке можна порівнювати значення полярних кутів за величиною без розрахунку їх точного значення. Це дозволяє отримати приріст швидкодії порівняно з швидкодією функції обчислення координат, що використовувалася раніше (Atan2 [6]).

Дана функція використана співавтором доповіді Семенякіним В.С. для реалізованої ним програми розрахунку областей видимості [1]. Під час реалізації

вказаної програми для пошуку *FOV* був проведений тест швидкодії отриманої функції.

При тесті продуктивності на комп'ютері з конфігурацією Intel Core2 Duo E4500, 1200 MHz зазначена функція, запущена десять мільйонів разів, виконувалася 1768 мс, у той час, як *Atan2*, що використовується в багатьох мовах програмування для точного розрахунку параметра [6] (в тесті, функція з STL, компіляція тестової програми під GCC 4.5.2) – 1857 мс. Маленький розрив за часом виконання пов'язаний з тим, що для процесорів лінійки x86 (що були випущені після 386 версії) існує апаратна реалізація функції *Atan* на математичному співпроцесорі.

У подальшому авторами планується провести порівняння швидкодії програмної реалізації функції *ACV* на асемблері з реалізацією функції *Atan2* і розробити комбінаційну схему для апаратної реалізації *ACV*. Також автори вважають за доцільне розробити математичний апарат для роботи із запропонованою евристичною функцією і дослідити можливість перенесення запропонованих ідей на тривимірний випадок (перехід із тривимірної декартової системи координат в сферичну і циліндричну системи).

Література

- [1] Field of view [Електронний ресурс/ http://en.wikipedia.org/wiki/Field_of_view], дата візиту 19.05.2011.
- [2] Полярная система координат [Електронний ресурс / http://ru.wikipedia.org/wiki/Полярная_система_координат], дата візиту 21.05.2011.
- [3] Основы компьютерной арифметики / Корнійчук В.І., Тарасенко В.П., Тарасенко-Клятченко О.В. – К.: «Корнійчук», 2006. – 164с.
- [4] Обратные тригонометрические функции: Разложение в бесконечные ряды [Електронний ресурс/http://ru.wikipedia.org/wiki/Обратные_тригонометрические_функции], дата візиту 20.05.2011.
- [5] [http://en.wikipedia.org/wiki/Sqrt\(\)](http://en.wikipedia.org/wiki/Sqrt()) [Електронний ресурс / [http://en.wikipedia.org/wiki/Sqrt\(\)](http://en.wikipedia.org/wiki/Sqrt())], дата візиту 20.05.2011
- [6] atan2 [Електронний ресурс / <http://en.wikipedia.org/wiki/Atan2>], дата візиту 20.05.2011.