

ПРО ДЕЯКІ МОЖЛИВОСТІ ЗАСТОСУВАННЯ ВЕКТОРНОЇ АЛГЕБРИ У ПРОЦЕСІ СТВОРЕННЯ КОМП'ЮТЕРНИХ ПРОГРАМ

Вінницький національний технічний університет

Анотація У статті розглянуто приклади застосування основних знань векторної алгебри при розробці комп'ютерних ігор. Для програмної реалізації переміщення віртуальних об'єктів необхідні знання про додавання (віднімання) векторів, скалярний та векторний добуток. Запропоновані приклади можна впроваджувати у навчальний процес з метою підвищення рівня мотивації студентів до вивчення розділу векторної алгебри.

Ключові слова: векторна алгебра, інтегративний підхід, комп'ютерна гра, мотивація.

Abstract. Examples of basic knowledge of vector algebra in computer games design are discussed in the article. Programmatic implementation of moving virtual objects requires knowledge of adding (subtracting) vectors, scalar and vector products. The offered examples can be implemented in the educational process in order to increase the students' motivation to study the section of vector algebra.

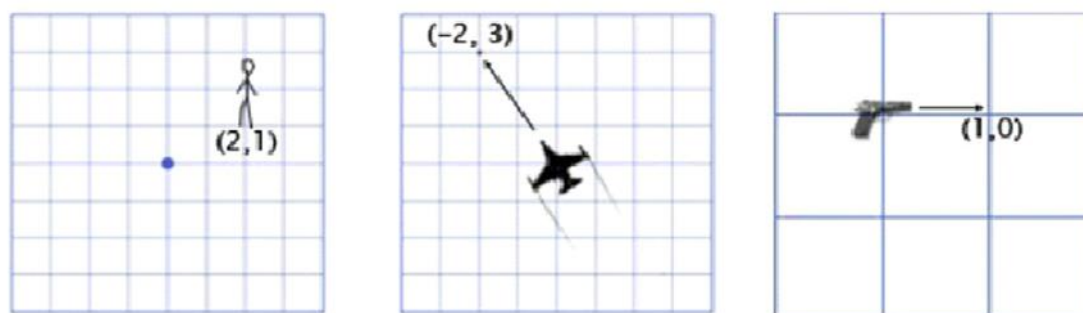
Keywords: vector algebra, integrative approach, computer game, motivation.

Вступ

Часто при вивченні розділів вищої математики студенти не розуміють прикладного значення інформації, що вивчається. Відповідно у студентів виникає питання "Навіщо вивчати матеріал, який не містить практичного застосування". Тому питання прикладного застосування отриманих знань є ключовим в навчальному процесі і обумовлює формування мотиваційної складової компетентнісного підходу. Доцільно для майбутніх інженерів наводити приклади, що містять прикладне застосування навчального матеріалу у контексті інтегративного підходу. Наведемо приклади застосування векторної алгебри в розробці комп'ютерних програм.

Результати дослідження

Векторна алгебра часто застосовується в розробці комп'ютерних програм та ігор. Проте творці програм сприймають вектор дещо інше від математиків. Розглянемо декілька прикладів



А) Положення

Б) Швидкість

В) Напрямок

Рисунок 1. Координати вектора в різних інтерпретаціях

Вектор положення (Рис 1., А) вказує на те, що чоловік стоїть два метри на схід від вихідного положення та один метр на північ. Умовно можна сказати, що вектор швидкості (Рис.1, Б) показує,

що за одну хвилину літак рухається на три кілометри вгору і на два вліво від заданої початкової точки. Вектор *напрямку* говорить нам, що пістолет спрямований праворуч.

Буквально, якщо не надавати вектору певного фізичного змісту, то розробники комп'ютерних ігор сприймають вектор, як набір чисел. Наприклад, вектор (1,0) може бути напрямком для пістолета, як показано на рисунку, але він також може бути положенням будівлі в кілометрі на схід від вашого поточного положення, або швидкістю равлика, що рухається праворуч зі швидкістю 1 км / год.

Нехай маємо вектор з координатами (3, 5, 2). Ці значення для програміста не несуть ніякого смислового навантаження в розробці програми. Тому для нього виникають питання “Три що?” “П'ять чого?” У грі *Overgrowth* позиції завжди задаються в метрах, а швидкості - в метрах за секунду. В заданих координатах вектора – положення (3, 5, 2) перше число – може визначати схід, друге - вгору, а третє - північ. Від'ємні числа представляють протилежні напрямки: захід, вниз та південь. Положення матеріальної точки, представлене (3, 5, 2), знаходиться на 3 метри на схід, на 5 метрів вгору і на 2 метри на північ, як показано на (рис 2) :

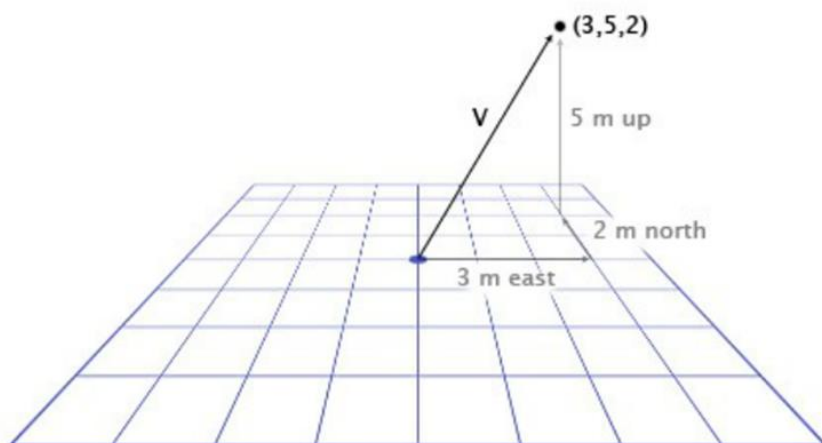


Рисунок 2. Задання вектором положення точки у просторі

ДОДАВАННЯ ВЕКТОРІВ

Загально відомо, щоб додати вектори, потрібно додати його відповідні координати. Так, $(0, 1, 4) + (3, -2, 5) = (0 + 3, 1 - 2, 4 + 5) = (3, -1, 9)$

Одне із найпоширеніших застосувань векторів в іграх у своїй основі передбачає інтеграцію з фізикою. Будь-який об'єкт, заснований на фізичній основі, швидше за все ілюструється через положення, швидкість та прискорення. Для кожного кадру (як правило, 1/60 секунди) для відображення руху, потрібно інтегрувати (сумувати) вектори – тобто додати швидкість до положення, а прискорення до швидкості.

Розглянемо гру Маріо. Маріо починає стрибати з позиції (0, 0). Коли він починає стрибок, його швидкість становить (1, 3) – він швидко рухається вгору, але також і вправо. Протягом цих дій його прискорення становить (0, -1), тому що на нього діє сила тяжіння, яка спрямована вниз. Ось як виглядає його стрибок протягом ще семи кадрів (Рис. 3). Чорний текст визначає його швидкість для кожного кадру.

Для першого кадру потрібно додати швидкість (1, 3) до його положення (0, 0), щоб отримати його нове положення (1, 3). Потім додаємо його прискорення (0, -1) до його швидкості (1, 3), щоб отримати його нову швидкість (1, 2).

Робимо це ще раз для другого кадру. Додаємо його швидкість (1, 2) до його положення (1, 3), щоб отримати (2, 5). Потім додаємо його прискорення (0, -1) до його швидкості (1, 2), щоб отримати (1,1).

Зазвичай в іграх гравець контролює прискорення персонажа за допомогою клавіатури або геймпада, і гра обчислює нову швидкість і положення (за допомогою додавання векторів).

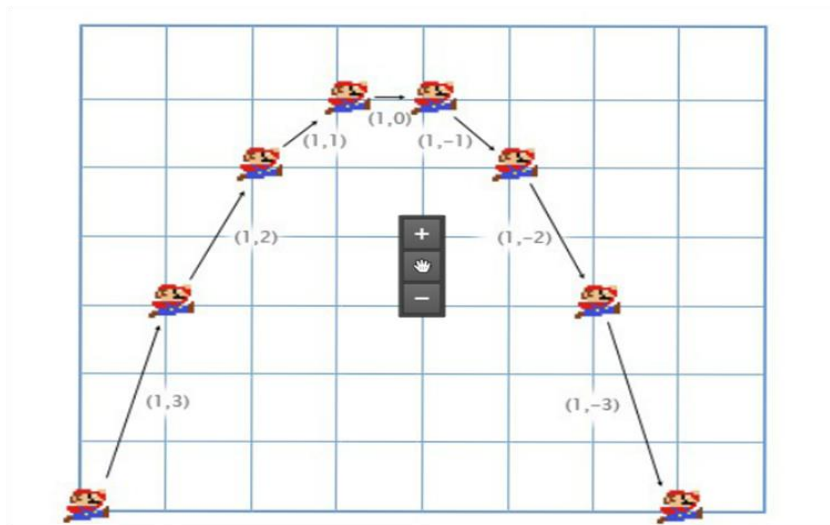


Рисунок 3. Векторне представлення руху Маріо

ВІДНІМАННЯ ВЕКТОРІВ

Віднімання векторів виконують аналогічно додаванню. Віднімання векторів доцільно застосовувати для отримання вектора, який напрямлений з однієї позиції на іншу. Наприклад, гравець стоїть на позиції (1, 2) з лазерною гвинтівкою, а ворожий робот - на позиції (4, 3) (Рис.4). Щоб отримати вектор, яку лазер повинен пройти, щоб потрапити на робота, потрібно відняти координати позиції гравця від координат позиції робота. Отримаємо: $(4, 3) - (1, 2) = (4-1, 3-2) = (3, 1)$.

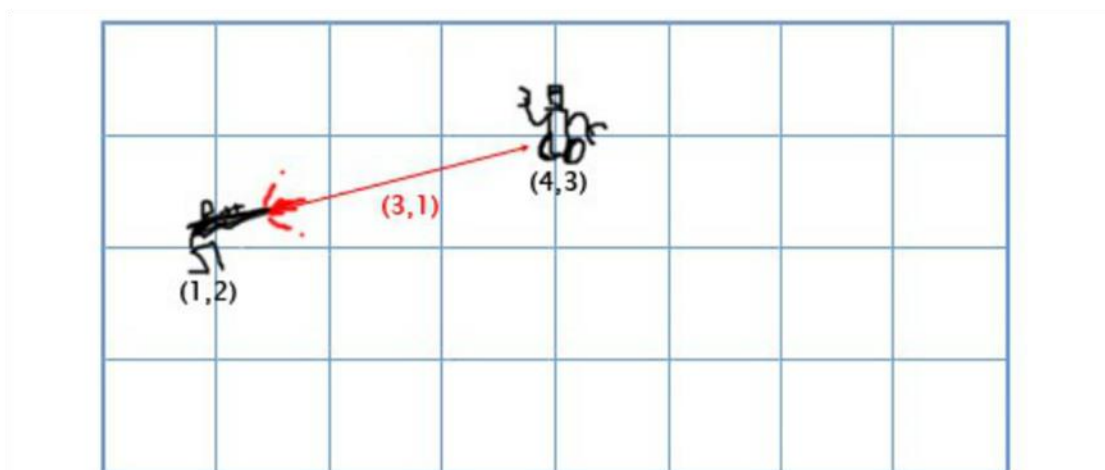


Рисунок 4. Ілюстрація застосування віднімання векторів в створенні ігор

СКАЛЯРНИЙ ДОБУТОК ВЕКТОРІВ

Приклад. Нехай у комп'ютерній грі є охоронець на позиції G (1, 3), спрямований в напрямку D (1, 1), з полем зору 180 градусів (Рис. 5). До нього підкрадається герой на позиції H (3, 2). Постає проблема:

чи буде він у полі зору охоронця? Ми можемо це з'ясувати, перевіривши знак скалярного добутку векторів D і V (вектор від охоронця до героя). Це дає нам:

$$V = H - G = (3, 2) - (1, 3) = (3-1, 2-3) = (2, -1)$$

$$D \cdot V = (1, 1) \cdot (2, -1) = 1 \cdot 2 + 1 \cdot (-1) = 2 - 1 = 1$$

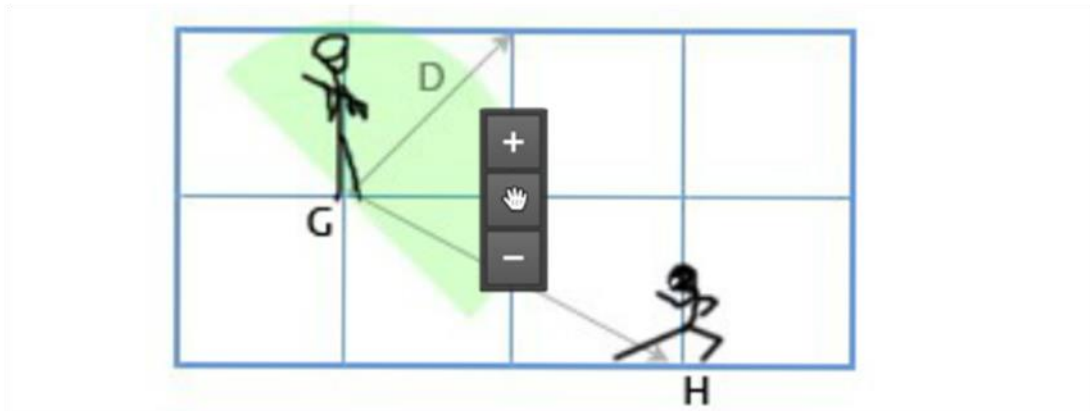


Рисунок 5. Ілюстрація застосування скалярного добутку векторів у розробці ігор

Оскільки, результат множення 1, що є додатнім числом, то можемо констатувати, що герой знаходиться у полі зору охоронця!

Отже, операція скалярного множення є застосовною при розробці ігор, особливо у випадках коли потрібно знати поле зору ігрового об'єкту і т.д. Якщо результат скалярного множення є додатнім числом то вектори вказують в одному напрямку, якщо число дорівнює нулю, то ці вектори є перпендикулярними, у випадку якщо число є від'ємним - вектори будуть спрямовані в протележному напрямку.

ВЕКТОРНИЙ ДОБУТОК

Розглянемо фрагмент комп'ютерної гри, у якому є корабель у 3D просторі. Є вектор напрямку мачти M , що прямує вгору $(0, 1, 0)$, і напрямку вітру (північ-північний схід) W $(1, 0, 2)$, і ми хочемо знайти напрямок паруса S , щоб найкраще піймати вітер. Парус має бути перпендикулярним до мачти, а також перпендикулярним до вітру.

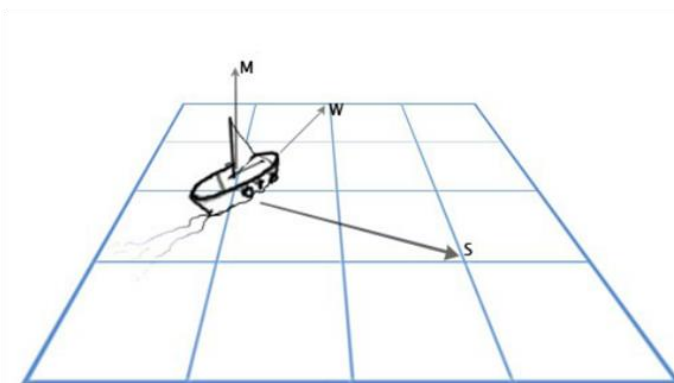


Рисунок 6. Фрагмент комп'ютерної гри

Для вирішення цієї проблем застосовують векторний добуток: $S = M \times W$.

Нагадаємо, що векторний добуток векторів $A(a_1, a_2, a_3)$ та $B(b_1, b_2, b_3)$:

$$A \times B = a_2 b_3 - a_3 b_2, a_3 b_1 - a_1 b_3, a_1 b_2 - a_2 b_1$$

Підставимо координати векторів з умови задачі:

$$S = M \times W = (0, 1, 0) \times (1, 0, 2) = ([1 \cdot 2 - 0 \cdot 0], [0 \cdot 1 - 0 \cdot 2], [0 \cdot 0 - 1 \cdot 1]) = (2, 0, -1)$$

Отже, ми можемо легко знайти напрям довільного об'єкта в грі

Висновки

У розробці комп'ютерних програм важливе місце посідає застосування векторної алгебри. Знання її теорії та вміння застосовувати ці знання для розробників комп'ютерних ігор є запорукою успішного створення програм. Для студентів першого курсу при вивченні розділу вищої математики "Векторна алгебра" доцільно демонструвати прикладне застосування одержаних знань. Що якісно підвищить рівень мотивації студентів до вивчення тем цього розділу.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Ключко В.І. Формування мотивації навчально-пізнавальної діяльності студентів технічних спеціальностей : монографія / В. І. Ключко, А. А. Коломієць. – Вінниця : ВНТУ, 2012. – 202 с.
2. Козловська І. М. Теоретико-методологічні аспекти інтеграції знань учнів професійно-технічної школи (дидактичні основи) / І. М. Козловська; ред.: С. У. Гончаренко; АПН України. Ін-т педагогіки і психології проф. освіти. - Л. : Світ, 1999. - 301 с.
3. Коломієць А.А. Інтегративний підхід в процесі формування змісту фундаментальної підготовки з математики майбутніх інженерів // Наукові записки. – Випуск 10. – Серія : Проблеми методики фізико-математичної і технологічної освіти. Ч.3 / За заг. ред. М. І. Садового. – Кропивницький : РВВ ЛДПУ ім. В. Винниченка, 2016 – 192 с., С.13-17.
4. Game Engine Architecture, Second Edition by Jason Gregory (ст 165 - 227).
5. <https://www.gamedev.net/articles/programming/math-and-physics/practical-use-of-vector-math-in-games-r2968/>

Коломієць Альона Анатоліївна - канд. пед. наук, доцент кафедри вищої математики, Вінницький національний технічний університет, м. Вінниця. alona.kolomiets.vny@gmail.com

Василевич Олександр Олегович - студент групи 1Ki -19 мс, Вінницький національний технічний університет, м. Вінниця. alexvlua@gmail.com

Kolomiets Alona A. — PhD Department of Higher mathematics, Vinnytsia National Technical University, Vinnitsa alona.kolomiets.vny@gmail.com

Vasilevich Alexander Olegovich - student group 1Ki -19 мс, Vinnytsia National Technical University Vinnitsa alexvlua@gmail.com