

МІНІСТЕРСТВО ОСВІТИ І НАУКИ,  
МОЛОДІ ТА СПОРТУ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ  
“ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ”

## МОДЕЛЮВАННЯ ФІЗИЧНИХ ЯВИЩ ІЗ ЗАСТОСУВАННЯМ ДИНАМІКИ ЧАСТИНОК

Методичні вказівки для лабораторних занять та  
самостійної роботи студентів із дисципліни  
«Моделювання спеціальних ефектів в комп’ютерній графіці»



Харків 2013

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ,  
МОЛОДІ ТА СПОРТУ УКРАЇНИ**

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ  
“ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ”**

**МОДЕЛЮВАННЯ ФІЗИЧНИХ ЯВИЩ  
ІЗ ЗАСТОСУВАННЯМ ДИНАМІКИ ЧАСТИНОК**

Методичні вказівки для лабораторних занять та  
самостійної роботи студентів із дисципліни  
«Моделювання спеціальних ефектів в комп'ютерній графіці»

Затверджено  
редакційно-видавничою  
радою університету,  
протокол № 2 від 01.12.10.

Харків  
НТУ«ХП»  
2013

**Моделювання фізичних явищ із застосуванням динаміки частинок**  
Методичні вказівки для лабораторних занять та самостійної роботи студентів із дисципліни «Моделювання спеціальних ефектів в комп'ютерній графіці» / уклад.: Савченко Л.М., Воронцова Д.В., Роженко З.М. – Харків: НТУ «ХПІ», 2013. – 52 с.

Укладачі: Л.М. Савченко,  
Д.В. Воронцова,  
З.М. Роженко

Рецензент      І.І. Сукачов, канд.техн.наук

Кафедра геометричного моделювання та комп'ютерної графіки

## ВСТУП

Комп'ютерна графіка широко використовується в різних сферах діяльності: від освіти, медицини, реклами і кіно до промислової індустрії та інженерної практики. Підготовка спеціалістів, що володіють методами геометричного моделювання в комп'ютерній графіці та анімації, ведеться на кафедрі геометричного моделювання та комп'ютерної графіки НТУ «ХПІ». Однією з цікавих, але дуже непростих задач, яка ставиться перед студентами, є створення спеціальних або, як їх ще називають, візуальних, чи цифрових, ефектів за допомогою комп'ютерної графіки.

Візуальні ефекти – це створені за допомогою комп'ютерної графіки такі види спецефектів як вогонь, дим, вибухи, дощ, сніг, блискавки та багато інших. При їх моделюванні найбільше поширення знайшли системи частинок. Частинка – це точка в тривимірному просторі, а група таких точок при візуалізації імітує сфери, хмари, бризки і т. ін. Змодельовані візуальні ефекти, з одного боку, мають бути реалістичними, а з другого – створювати у спостерігача неперевершене враження. Для отримання різноманітних візуальних ефектів використовують системи частинок, в яких за допомогою математичних функцій контролюється розмір частинок, їхній колір, прозорість, динамічні властивості. У кожній частинки є набір індивідуальних атрибутів, приналежних тільки їй (положення, швидкість, маса, прискорення, тривалість існування). Решту атрибутів, які надають до частинок додатково, називають динамічними. Їх можна додавати або до всіх частинок одразу, або для кожної окремо.

Вплив фізичних сил на частинки здійснюється за допомогою полів, які примушують їх рухатися в заданому напрямку (вплив вітру), або роблять переміщення частинок хаотичним, або закручують в спіраль, притягують до центру чи, навпаки, відштовхують від центра, імітують силу гравітації чи щось інше.

Метою даних методичних вказівок є знайомство студентів з методами створення систем частинок, вивчення їхніх основних властивостей, застосування впливу полів для отримання бажаного ефекту.

## ПРИРОДА ЧАСТИНОК

Частинки – це дуже дрібненькі тверді тіла, настільки дрібні, що в них відсутня поверхня, тобто це нескінченно малі, але важкі точки. Через відсутність поверхні вони нездатні зіштовхуватися між собою і коли знаходяться в одній точці простору, то можуть «пролітати» одна крізь одну. Саме відсутність взаємних зіткнень, і як наслідок, необхідність у прорахунках цих зіткнень дозволяє робити інтерактивну анімацію тисяч частинок. Частинки можуть також стикатися з перепонами і відскакувати від них із заданою пружністю. При цьому вони можуть зникати, розпадатися на шматки або випускати нові частинки. Вони мають початкову швидкість і такі атрибути, які при редагуванні в кожному кадрі можуть керувати положенням, швидкістю та прискоренням. Для частинок можна задати ціль, до якої вони будуть прямувати (летіти). Роботу з частинками умовно розбивають на три етапи:

- створення системи частинок;
- моделювання бажаного ефекту та анімація;
- візуалізація.

## СТВОРЕННЯ ЧАСТИНОК

Існує кілька способів створення частинок. Насамперед, їх можна намалювати вручну за допомогою **Particles** → **Particle Tool**. В **Option Box** для **Particles** → **Particle Tool** є настройки, які дозволяють створювати відразу сітку (рис.1)

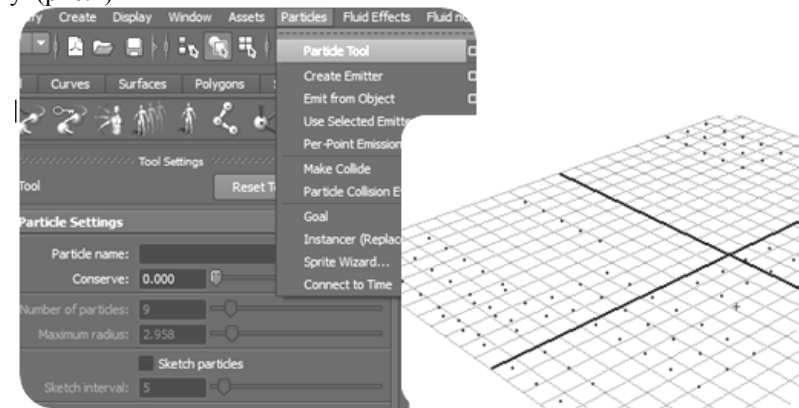


Рисунок 1 – Створення частинок за допомогою інструмента **Particle Tool**

В розділі *Render Attributes* вікна *Attribute Editor* натисніть на кнопку *Add Attributes for Current Render Type* (дати атрибути для поточного типу візуалізації), щоб отримати доступ до редагування додаткових атрибутів. Наприклад, щоб змінити розмір частинки, треба внести зміни в значення параметру *Point Size* (Розмір точки) рис.2.

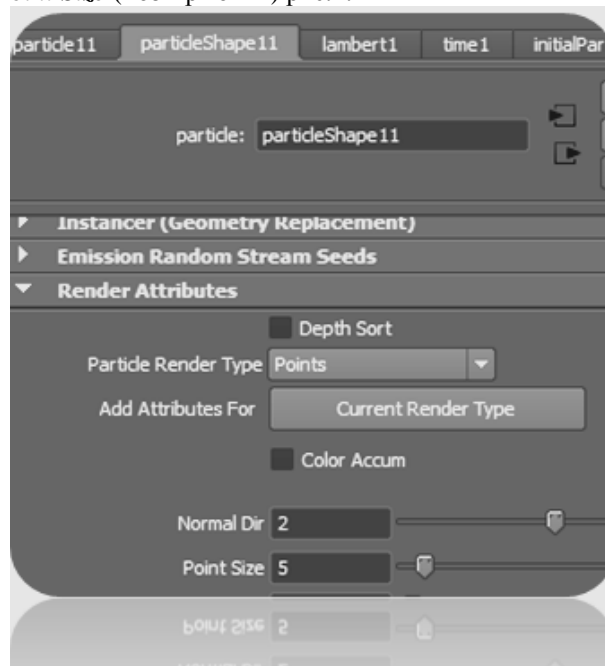


Рисунок 2 – Редагування додаткових атрибутів частинок

**ЗВЕРНИ УВАГУ!** Частинки можна малювати у вікні проєкцій. Для цього треба активізувати команду *Sketch Particles* (ескіз) в настройках інструмента *Particle*. За допомогою параметрів *Number of particles*, *Sketch interval*, *Maximum radius* можна впливати на кількість частинок, що одночасно з'являються, інтервал між ними та ширину полоси їх розповсюдження (рис.3).

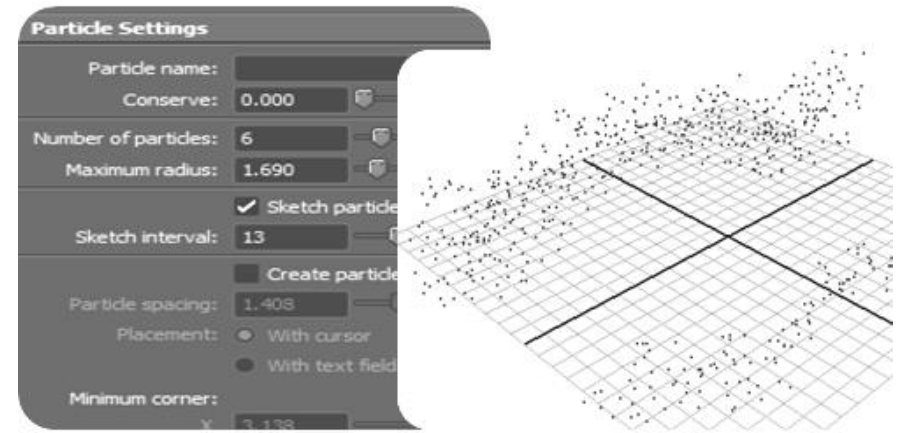


Рисунок 3 – Створення системи частинок за допомогою інструмента *Sketch Particles*

Однак частіше частинки створюють не вручну, а за допомогою джерела частинок (емітера), який розпиляє їх із заданою швидкістю і в заданому напрямку.

Найпростішим джерелом частинок є *точковий емітер* (або *генератор*). Його створюють за допомогою команди *Particles* → *From Emitter*. Параметри емітера задають через *Option Box* (рис.4, а) або *Channel Box* чи *Attribute Editor*. Існують три типи емітерів:

- *Omni* (випускає частинки у всіх напрямках),
- *Directional* (розпиляє частинки в одному напрямку),
- *Volume* (випускає частинки з об'ємної простої форми).

Тип *Volume* перетворює точковий емітер в об'ємний і дозволяє задавати кілька примітивних форм (куб, циліндр, сферу, конус та тор) рис.5.

**ЗВЕРНИ УВАГУ!** При виконанні операції *Create Emitter* створюються два об'єкта: безпосередньо джерело і частинки, що вилітають з нього. Частинки являють собою повністю незалежний об'єкт і не входять до складу компонентів емітера, але вони приєднуються до нього.



а

б

Рисунок 4 – *Option Box* емітера частинок (а), та основні його атрибути (б)

#### АТРИБУТИ ГЕНЕРАТОРА ЧАСТИНОК

- **Rate (particle/sec)** (частота) – визначає число частинок, що випускаються генератором за одну секунду;
- **Speed** (швидкість) – швидкість частинок;
- **Speed Random** (випадкова швидкість) – визначає границі діапазону, в яких змінюється швидкість частинок;
- **Min Distance** (мінімальна відстань) / **Max Distance** (максимальна відстань) – задають відстань від генератора, на якій створюються частинки (рис.4, б).



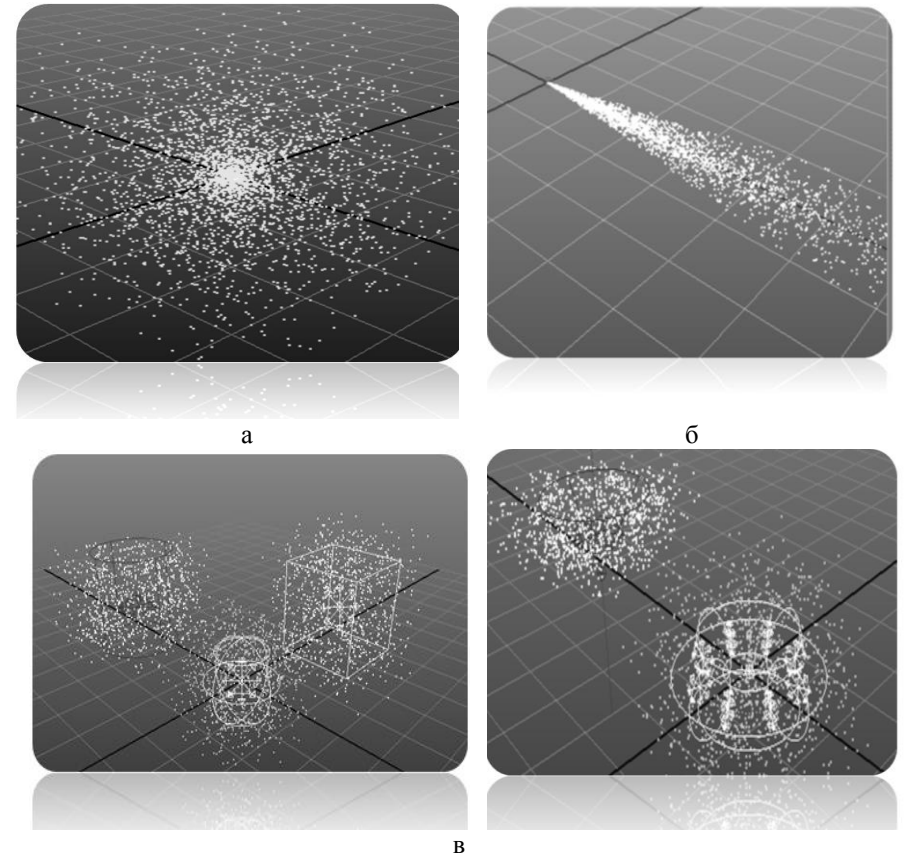


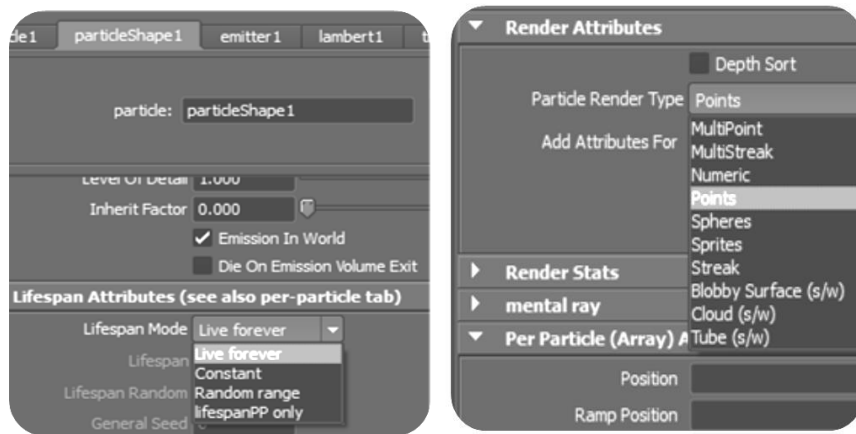
Рисунок 5 – Частинки, які створені емітерами *Omni* (а), *Directional* (б), *Volume* (в)

#### АТРИБУТИ ЧАСТИНОК

Для виділеної системи частинок в вікні діалогу *Attribute Editor* на вкладці *Particle 1* розміщені атрибути трансформацій: положення, повороту та масштабування. Параметри форми розташовані на вкладці *Particle Shape 1*. Вкладка *emitter 1* містить параметри генератора частинок.

Одним із основних параметрів частинок є час перебування їх в сцені.

При моделюванні різноманітних ефектів виникає необхідність обмежувати період існування частинок або змінювати деякі властивості на протязі визначеного інтервалу часу. Вибрати режим часу існування можна із списку **Lifespan mode** (рис.6). За замовчанням обирається варіант **Live Forever** (нескінчений час життя). Це означає, що частинки які з'явилися, залишаються назавжди. Режим **Constant** (постійне існування) дозволяє задавати проміжок часу існування частинок, що вказується в полі **Lifespan** (час існування), після якого вони зникають. Режим **Random Range** (випадковий діапазон) урізноманітнює період існування частинок. Атрибут **Lifespan Random** (випадковий час існування) вказує діапазон, в якому може змінюватися час життя частинок. Вибір режиму **Lifespan PP Only** (час життя однієї частинки) дає можливість контролювати період існування окремих частинок, яке можна задавати за допомогою мови **MEL**.



а

б

Рисунок 6 – Атрибути частинок: а – період існування; б – типи частинок та способи їхньої візуалізації

Параметри вкладки **Render Attributes** (атрибути візуалізації) визначають тип частинок і спосіб їхньої візуалізації: програмний або апаратний. Апаратна візуалізація частинок здійснюється окремо від основної сцени і поєднується з нею на стадії монтажу. В розділі **Render Attributes** знаходиться

список **Particle Render Type** (тип візуалізації частинок), в якому три варіанта частинок супроводжуються написом *s/w*, що вказує на програмну візуалізацію частинок. Решта візуалізуються апаратно (рис.6, б).

На рис.7 – 9 наведені *типи апаратно візуалізованих частинок*.

**Points** (точки) (рис.7,а) візуалізуються в вигляді точок, радіус яких можна змінювати.

**Multi Points** (мультиточки) (рис.7,б) виглядають як кластери точок. Атрибут **Multi Radius** контролює розмір кожного кластера, а **Multi Count** задає кількість точок в кожному кластері.

**Streak** (штрихи) (рис.8,а). Частинки візуалізуються в вигляді ліній, на вид яких можна впливати, змінюючи атрибути: **Size** (розмір), **Line Width** (ширина лінії), **Tail Size** (розмір сліду), **Tail Fade** (загасання сліду).

**Multi Streak** (мультиштрихи) – рис.8,б візуалізуються у вигляді ліній. Окрім властивостей **Streak**, цей тип візуалізації має атрибути **Multi Radius** і **Multi Count**.

**Sprites** (спрайти) – рис.9,а. Вид цих частинок задає користувач, призначаючи їм будь-який файл зображення із меню **Particle** → **Sprites Wizard**.

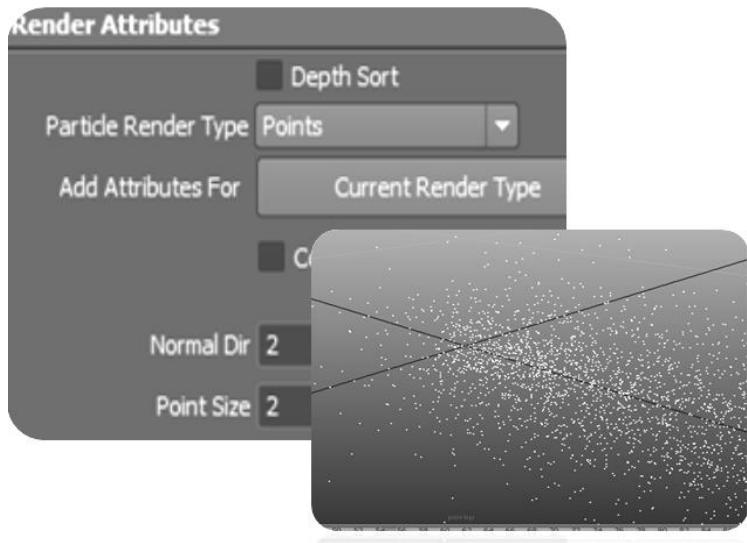
**Spheres** (сфери) – рис.9,б. Частинки візуалізуються в вигляді непрозорих сфер, яким можна призначити будь-який матеріал, але параметр «прозорість» не підтримується.

*Програмно візуалізовані частинки* поділяються на три типи: (рис.10 –12)

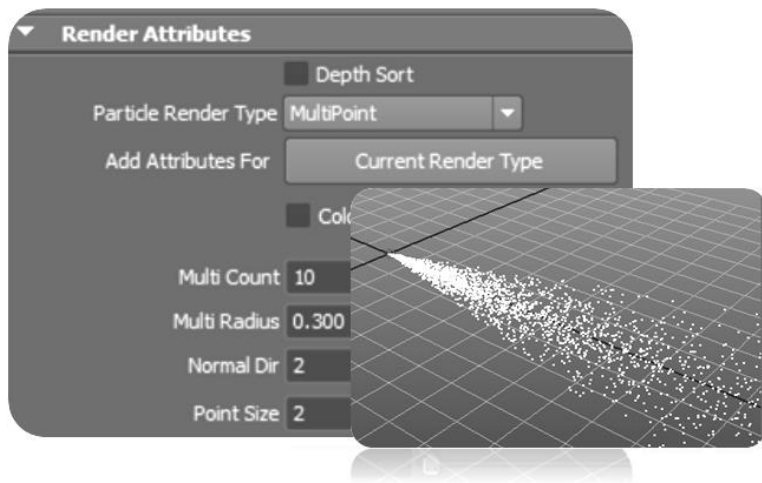
**Blobby Surface** (крапельна поверхня). Частинки набувають сферичної форми і злипаються одна з одною (рис.10). Їхні атрибути: **Radius** (радіус) – задає початкові розміри частинок; **Threshold** (поріг) – визначає ступінь взаємного проникнення частинок.

**Cloud** (хмара) – частинки набувають сферичної форми і стають більш прозорими при віддаленні від центру (рис.11), їхні властивості аналогічні атрибутам **Blobby Surface**, але при тонуванні обирають **Particle Cloud**.

**Tube** (труба) – частинки мають циліндричну форму і збільшуються при анімації. Атрибути **Radius0**, **Radius1** задають початкове та кінцеве значення збільшення, а атрибут **Tail Size** визначає довжину труби (рис.12).

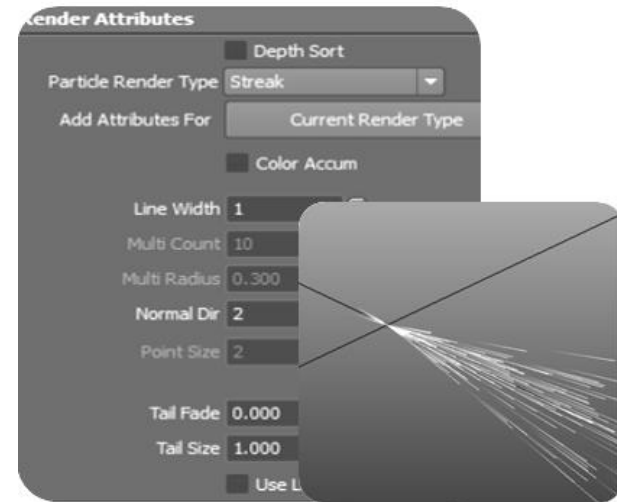


a

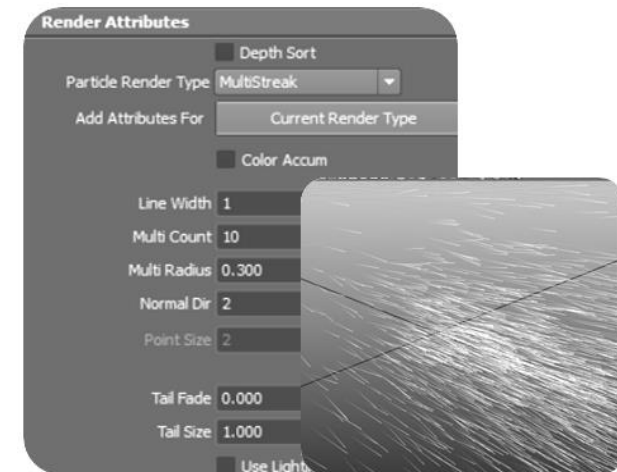


б

Рисунок 7 – типи апаратно візуалізованих частинок *Points* (а), та *Multi Points* (б)

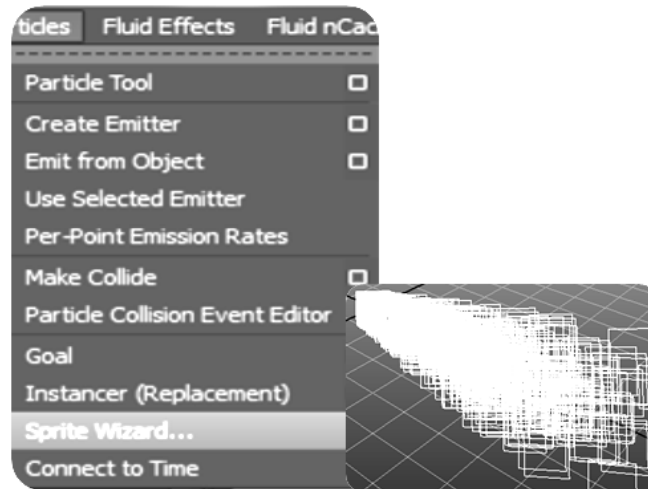


а

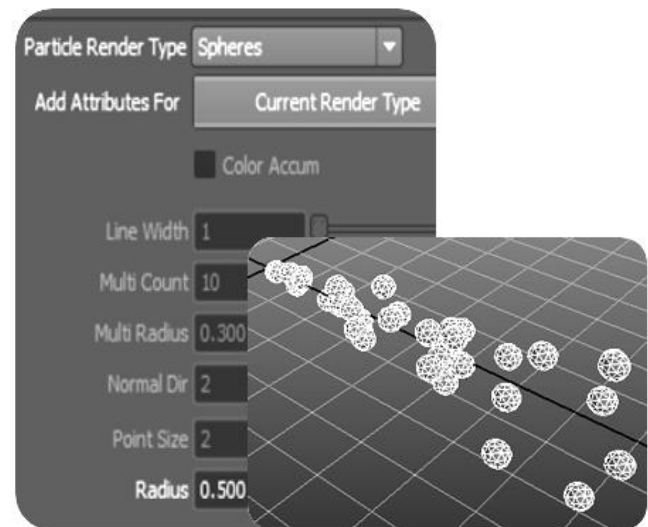


б

Рисунок 8 – Типи апаратно візуалізованих частинок **Streak** (а), та **Multi Streak** (б)



a



б

Рисунок 9 – Типи апаратно візуалізованих частинок *Sprites*(а) та *Sphere* (б)

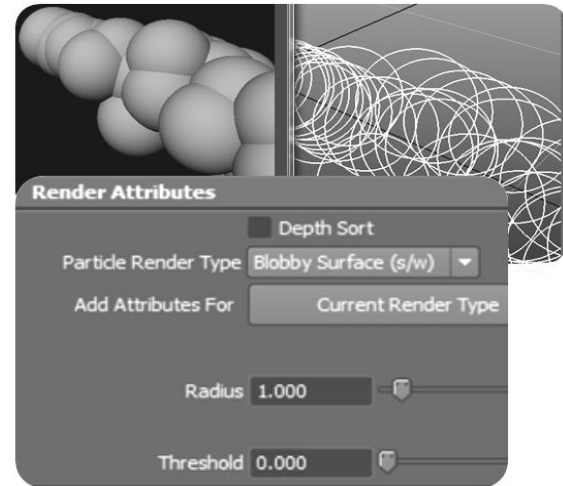


Рисунок 10 – Програмно візуалізовані частинки *Bobby Surface*

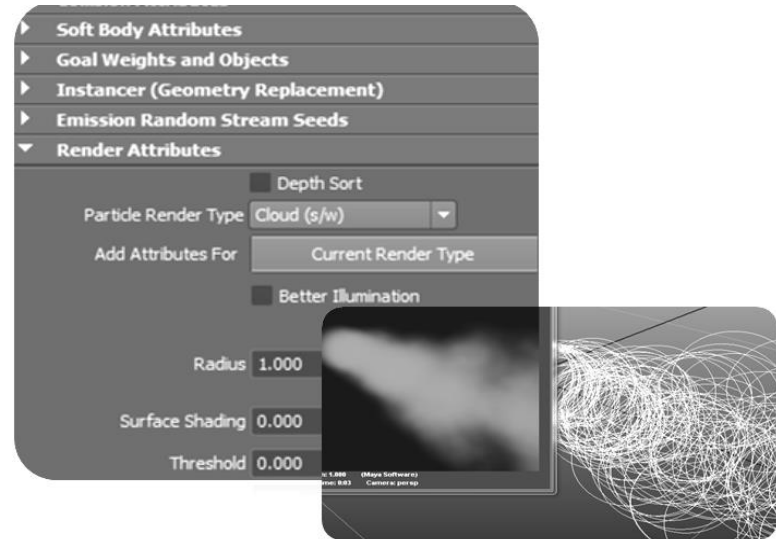


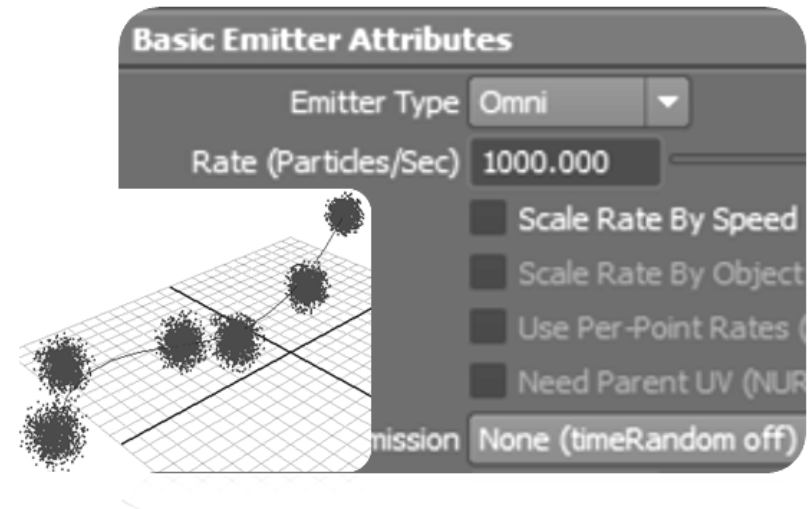
Рисунок 11 – Програмно візуалізовані частинки *Cloud*



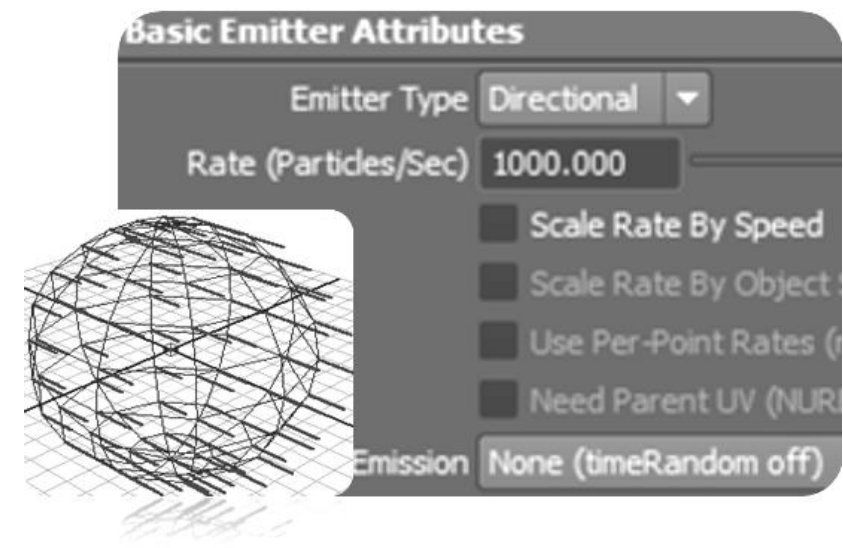
Рисунок 12 – Програмно візуалізовані частинки *Tube*

Крім описаних способів створення частинок, існує можливість випускати їх безпосередньо з поверхні або кривої. Джерелом частинок в цьому випадку стає сама поверхня. Створюється об'єкт типу емітер, який має додаткові сполучення з поверхнею, через які йде інформація про геометричну форму об'єкта, що випускає частинки точно з поверхні. Якщо створити криву або будь-яку поверхню і виконати команду *Particles → Emit from Object*, то частинки будуть випускатися не з поверхні, а із точок – компонентів поверхні за умови, що тип емітера встановлений *Omni* або *Directional* (рис.13).





a



б

Рисунок 13 – Випускання частинок з точок кривої або поверхні

Якщо установити *emitter Type Surface* чи *emitter Type Curve*, то частинки випускатимуться рівномірно з усієї поверхні чи кривої (рис.14). За замовчанням, кількість частинок, що вилітають за одиницю часу, не залежить від площі поверхні. Однак, якщо додати атрибут *Scale Rate By Object Size*, то значення *Rate* буде визначати кількість частинок, що вилітають з одиниці площі, а з усієї поверхні буде випускатися число частинок, яке дорівнюватиме добутку *Rate* на площу поверхні (рис.15).

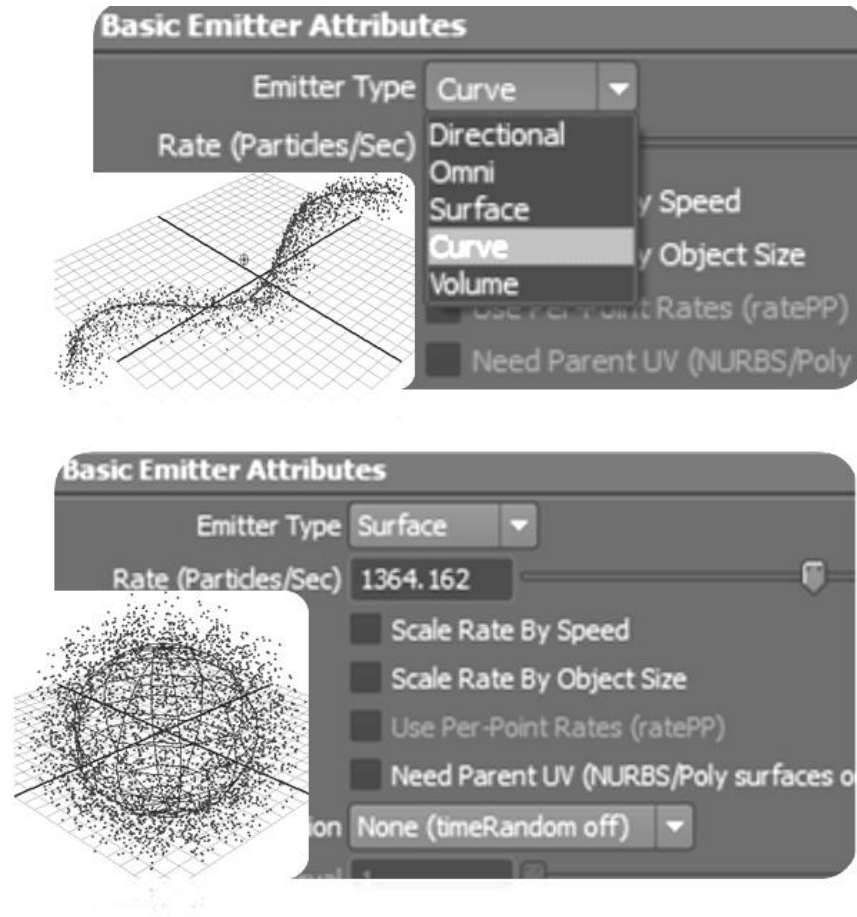


Рисунок 14 – Випускання частинок з кривої або поверхні

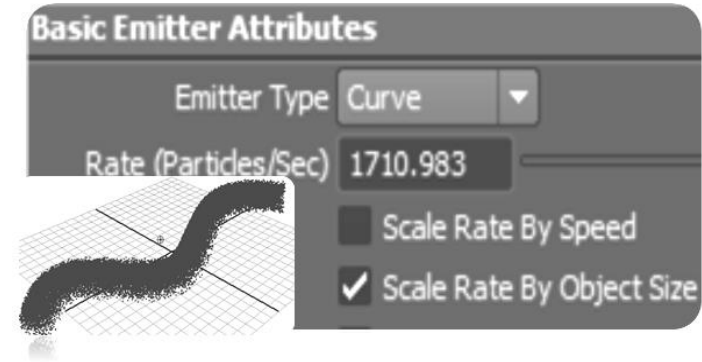


Рисунок 15 – Випускання частинок зі збільшеною щільністю з кривої або поверхні

Якщо треба змінити напрямок руху частинок на протилежний, то атрибут *normal Speed* встановлюють рівним -1. Якщо встановити *tangent Speed*=1, а *normal Speed* =0, то частинки будуть випускатися вздовж поверхні (рис.16).

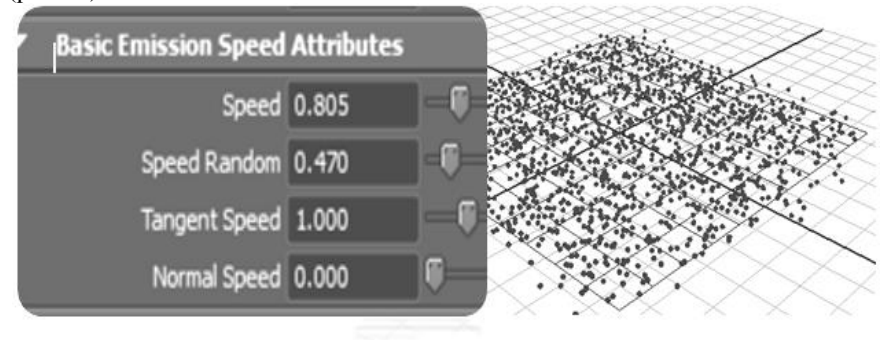


Рисунок 16– Випускання частинок вздовж поверхні

Щільність випускання частинок залежить від кольору, а точніше від текстури, яка задає області випускання з різною щільністю, і дозволяє це зробити атрибут *Texture Rate*, який знаходиться в розділі *Texture Emission Attributes* (рис.17). Щоб призначити текстуру, від якої б залежала щільність

випускання частинок, треба натиснути на шашку праворуч від *Texture Rate*, і в вікні, яке з'явиться, вибрати необхідну текстуру, наприклад, *Checker*. Включена галка для *Enable Texture Rate* додає різну щільність частинкам, що випускаються (рис.18).

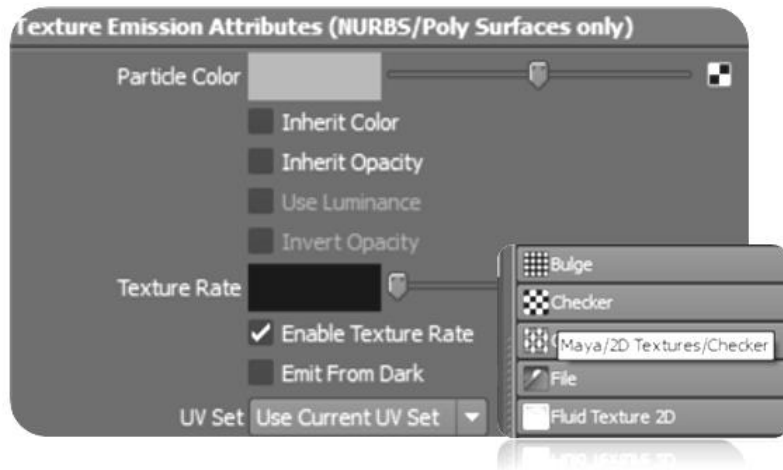


Рисунок 17 – Вплив текстури на щільність випускання частинок

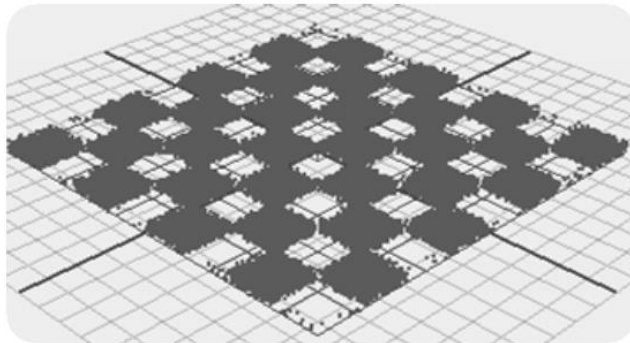


Рисунок 18 – Додавання текстури *Checker* до атрибуту *Texture Rate* для зміни щільності випускання частинок

Для того, щоб частинки мали колір, який відповідає обраній текстурі, треба включити галку **Inherit Color** (спадкувати колір) і натиснути на шашку праворуч від **Particle Color**, щоб додати текстуру на цей атрибут (при цьому не забути видалити галку для **Enable Texture Rate** рис.19). Необхідну текстуру вибрати із вікна, що з'явилося (наприклад, із текстури типа **File**) і для неї в полі **Image Name** завантажити будь-яке растрове зображення (рис.20), а далі натиснути на кнопку **Color** (рис.21) в розділі **Add Dynamic Attribute**. Тоді з'явиться діалогове вікно, в якому вибирають другий рядок, оскільки колір треба додати кожній частинці, потім натиснути на кнопку **Add Attribute**. Кожна частинка, що утворюється, має свій особистий колір, успадкований нею заданою текстурою (рис.22).

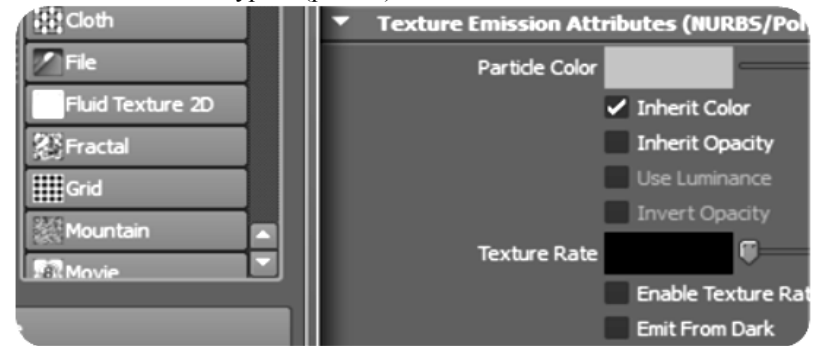


Рисунок 19 – Додавання кольору частинкам, що випускаються



Рисунок 20 – Рисунок із файла для додавання кольору частинкам

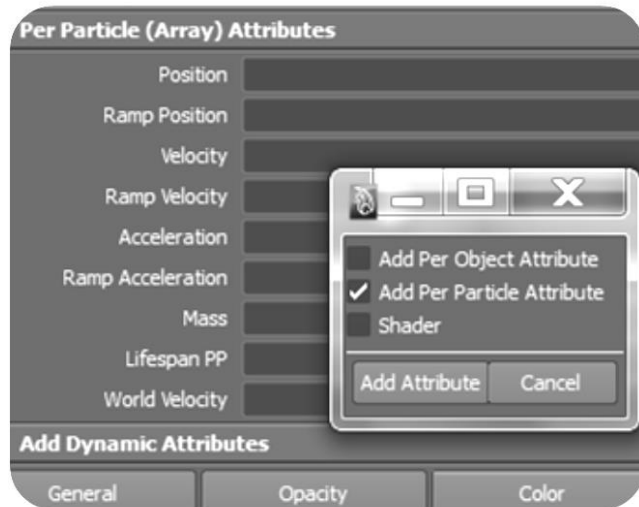


Рисунок 21 – Додавання кольору кожній частинці

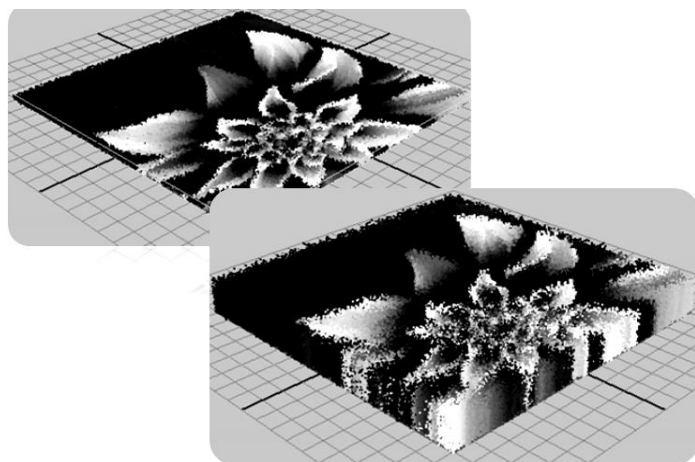


Рисунок 22 – Частинки, що успадкували заданий колір за допомогою обраної текстури

## СИЛОВІ ПОЛЯ

В дійсності частинки не є незалежні об'єкти, тому що на них діють різноманітні зовнішні сили, наприклад, гравітація та турбулентність, що призводить до зміни траєкторії їхнього руху. Створення силових дій забезпечують силові поля **Fields** (поля), що доступні в режимі **Dynamics** (динаміка) (рис. 23).

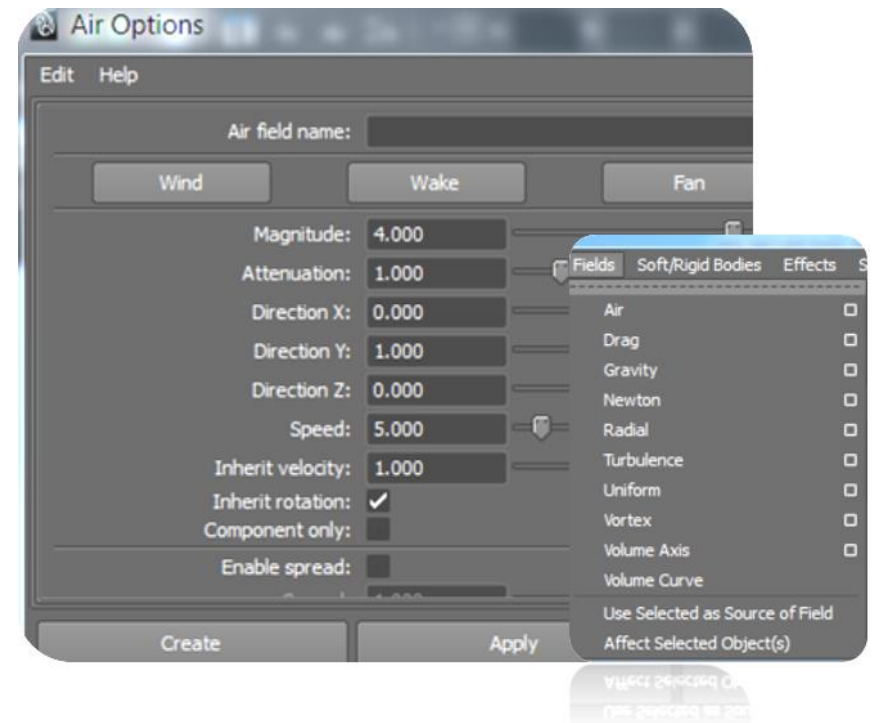


Рисунок 23 – Силові поля

На частинки можуть діяти наступні силові поля:

**Air** (вітер) — імітує дію вітру, що відхиляє частинки;

**Drag** (тертя) — уповільнює рух частинок внаслідок тертя;

**Gravity** (гравітація) — використовується для імітації впливу сили тяжіння;

**Newton** (сила Ньютона) — забезпечує орбітальне обертання частинок;

**Radial** (радіальність) — призводить до розкидання частинок в різні сторони або, навпаки, до притягування їх до центру;

**Uniform** (однорідність) – створює ефект подібний **Air**, виштовхує частинки в заданому напрямку;

**Turbulence** (турбулентність) — призводить до перемішування частинок під час руху;

**Vortex** (вихор) — створює ефект закручування частинок по спіралі;

**Volume Axis** (об'ємна вісь) – діє всередині будь-якої об'ємної форми.

Для поля **Air** існує ще три типи поля (рис.23):

**Wind** (вітер) – переміщує частинки в заданому напрямку, доти вони не наберуть визначену швидкість;

**Wake** (струміль) – створює ефект струменя, порушуючи структуру групи частинок і затягує їх за собою;

**Fan** (вентилятор) – створює ефект руху повітря від вентилятора, розштовхуючи частинки під кутом 45°.

#### ЗАГАЛЬНІ АТРИБУТИ ПОЛІВ

Для всіх типів полів характерні такі атрибути:

**Magnitude** (амплітуда) – задає величину сили, з якою поле діє на частинки;

**Attenuation** (затухання) – визначає коефіцієнт послаблення сили по мірі віддалення частинок від центра поля;

**Direction** (напрямок) – встановлює напрямок сили, зазвичай вздовж осі **X, Y, Z**;

**Use Max Distance** (використати максимальну відстань) – дозволяє задати найбільшу відстань області дії поля;

**Max Distance** (максимальна відстань) – задає відстань, на якій буде діяти сила поля.

Атрибути полів можна анімувати, тоді в залежності від часу ефект впливу параметрів полів на частинки буде змінюватися. Щоб додати дію поля до частинок треба їх виділити, далі вибрати з меню **Fields** необхідний тип поля. На рис.24 продемонстровано ефект закручування по спіралі частинок під впливом дії силового поля **Vortex**.



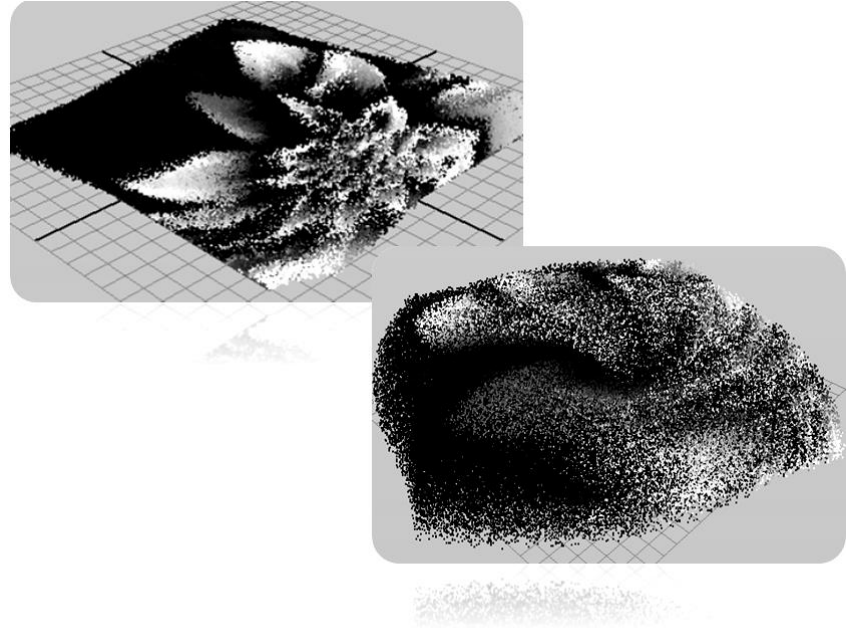


Рисунок 24 – Приклад впливу силового поля *Vortex* на частинки

### ЗІТКНЕННЯ ЧАСТИНОК З ПЕРЕПОНАМИ

Частинки можуть зіштовхуватися з будь-якими полігональними чи сплайновими об'єктами, окрім *NURBS*-кривих та одна з одною. При зіткненні вони відскакують від поверхні, втрачаючи при цьому швидкість, і можуть випускати нові частинки, або розбиватися на додаткові частинки, або просто зникати. Щоб призначити частинкам зіткнення, треба вибрати команду *Make Collide* (задіяти зіткнення) із меню *Particles* (рис.25).

Видалити зіткнення можна через редактор динамічних зв'язків *Dynamic Relations*, що викликається командою *Window* → *Relationship Editors* → *Dynamic Relations* (вікно → редактори зв'язків → редактор динамічних зв'язків) – рис.26.

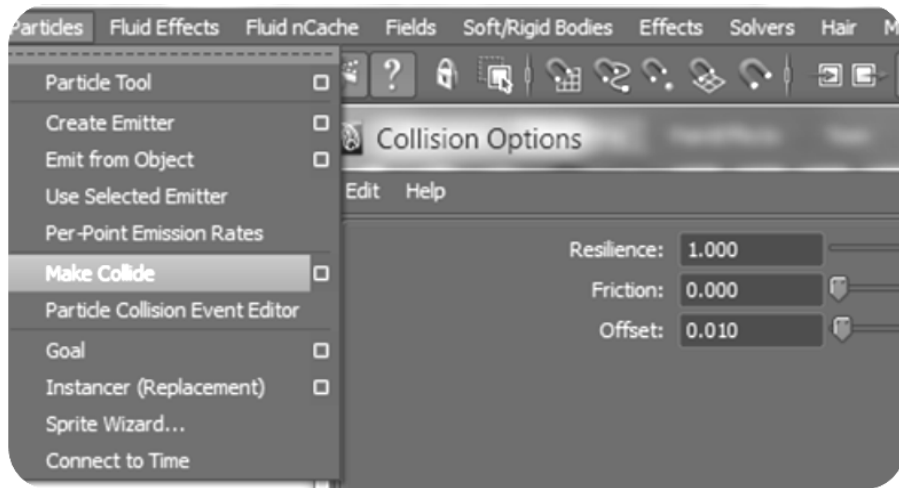


Рисунок 25 – Атрибути зіткнення частинок

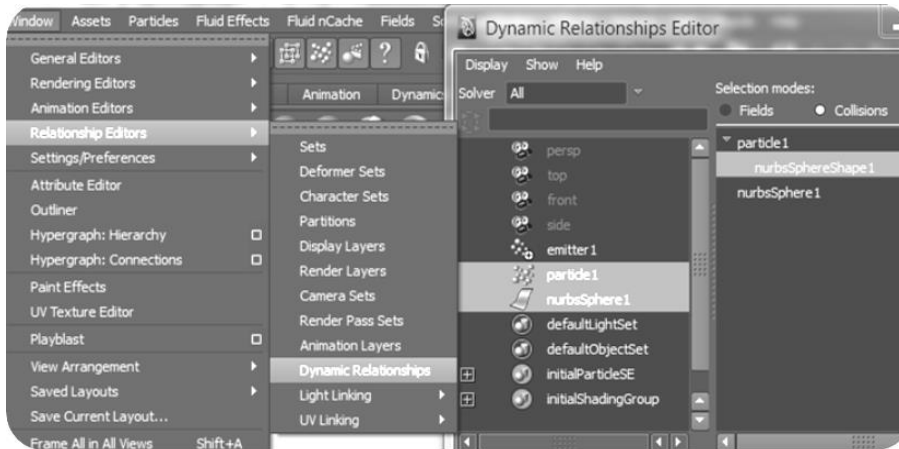


Рисунок 26 – Редактор динамічних зв'язків

Щоб видалити створене зіткнення, треба в лівій частині редактора динамічних зв'язків *Dynamic Relations* виділити систему частинок, а в правій встановити режим *Collisions*, який відповідає за управління зіткненнями. Після цього клацнути на рядку, що світиться іменем об'єкта, з яким зіштовхуються частинки і зв'язок буде порушеним. Якщо треба знову встановити зіткнення між системою частинок та об'єктом, то для цього достатньо буде лише клацнути на тому ж самому рядку в редакторі *Dynamic Relations* для його підсвічування. Особливостями зіткнення частинок можна управляти за допомогою параметрів *Resilience* (пружність) і *Friction* (тертя) — їх значення встановлюються відразу ж при налаштуванні параметрів команди *MakeCollide* або редагуються пізніше через редактор атрибутів на вкладці *geo Connector*. Параметр *Resilience* характеризує ступінь пружності частинок і зазвичай змінюється в інтервалі від 0 до 1: при значенні 0 частинки просто зіштовхуються з поверхнею, а при 1 — відлітають після зіткнення з початковою швидкістю. Параметр *Friction* визначає рівень тертя. Якщо значення параметра *Friction* дорівнює 0, то поверхня настільки гладенька, що частинки при зіткненні з нею взагалі не зазнають впливу тертя, а при значенні рівному 1, тертя буде максимальним, що спричинить зупинку частинок. На рис.27 показано частинки, які після зіткнення з перепonoю відлітають від поверхні сфери, з якою вони зіштовхнулися.

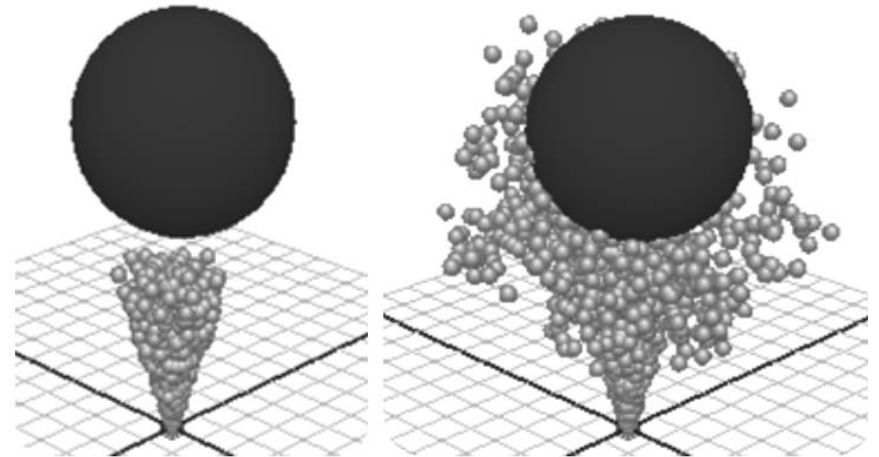


Рисунок 27 – Зіткнення частинок з перепonoю

Трапляються випадки, коли деякі частинки, які мали первісно високу швидкість переміщення, при повторних зіткненнях з деякою поверхнею, все одно пролітають крізь неї. Щоб завадити цьому треба в редакторі атрибутів на вкладці *particle Shape1* в секції *Collision Attributes* збільшити значення параметра *Trace Depth*, який відповідає за рівень виявлення зіткнень, до відповідного значення (рис 28).

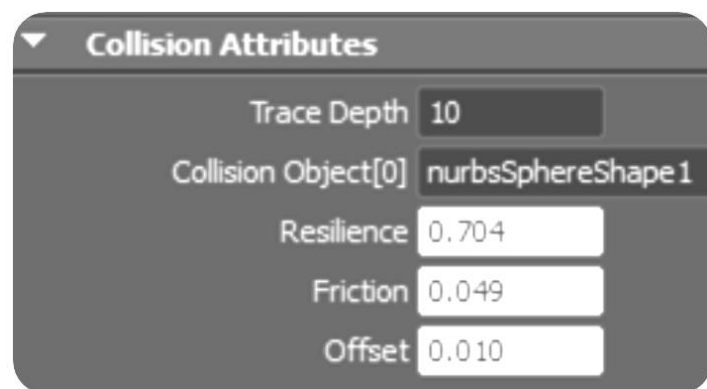


Рисунок 28 – Параметр *Trace Depth*, який відповідає за рівень виявлення зіткнень частинок з перепорою

Як уже відзначалось, після зіткнення частинок з поверхнею можливі такі варіанти розвитку подій: найчастіше всього після зіткнення частинки або стають джерелом нових частинок, або розбиваються на множину додаткових частинок, або просто помирають. Регулюється це через редактор подій частинок *Particle Collision Events Editor*, який вибирають командою *Particles*→*Particle Collision Events Editor* (рис. 29). В ньому вказують, для яких частинок даної системи частинок треба виконати подію:

*AllCollisions* — подія буде спрацьовувати для всіх частинок, що зіткнулися з перепорою, незалежно від номера зіткнення;

*Collision Number* — подія буде виконуватися тільки для частинок, які зіткнулися з перепорою вказану в даному параметрі кількість разів.

Крім того, визначається тип події: *Emit* або *Split*. Подія типу *Emit* означає, що частинки при зіткненні будуть випускати нові частинки і продовжувати рухатися, а в випадку події типу *Split* частинки при зіткненні розколюються на зазначену кількість додаткових частинок і помирають. Для обох типів подій передбачені настройки наступних параметрів:

**Random # Particles** — забезпечує генерацію випадкового числа частинок, яке може мінятися в інтервалі від 1 до значення, вказаного в параметрі *Num particles*;

**Num particles** — задає кількість нових частинок, що формуються (шляхом генерації або розколу) кожною із частинок, що зіткнулися;

**Spread** — визначає кут розкидання нових частинок;

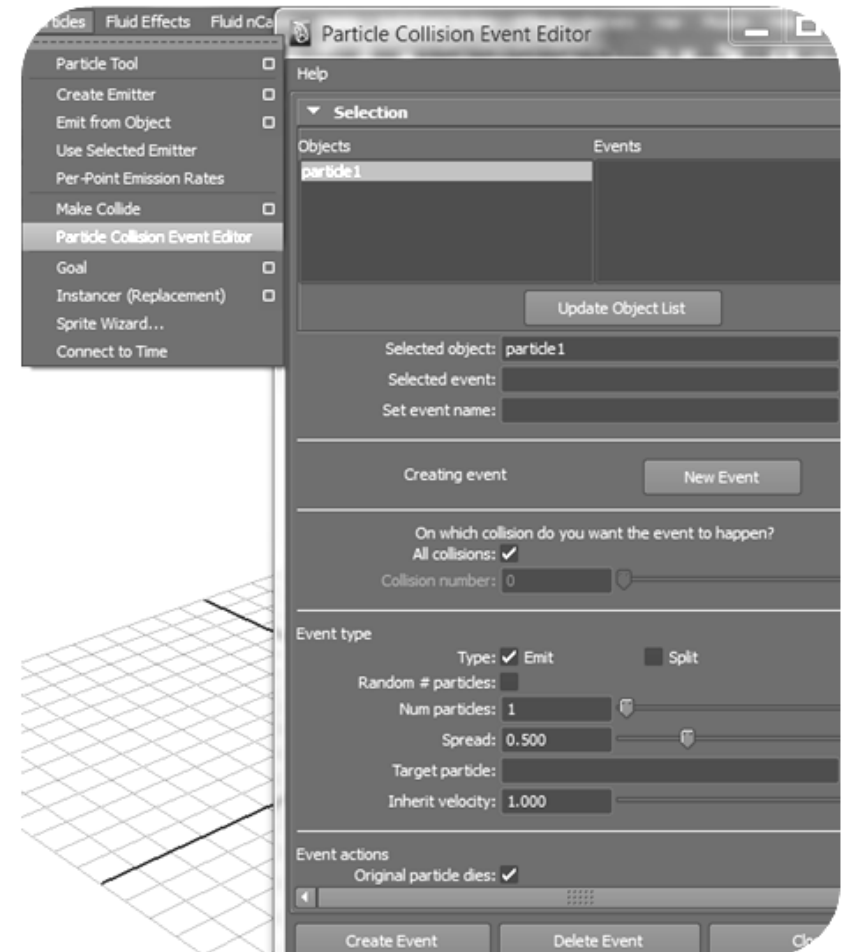


Рисунок 29 – Вікно редактора зіткнень *Particle Collision Events Editor*

**Target Particle** — вказує систему частинок (це може бути як нова, так і та, що вже існує), часткою якої стають згенеровані при зіткненні частинки;

**Inherit Velocity** — визначає швидкість нових частинок, яка задається як доля від швидкості частинок, що їх створили; даний параметр може змінюватися в інтервалі від 0 до 1: при 0 нові частинки не успадковують швидкості старих, при 1 — швидкості нових і старих частинок рівні.

Якщо треба видалити створену подію, достатньо відкрити редактор **Particle Collision Events**, виділити потрібну систему частинок, далі виділити подію і натиснути на кнопку **Delete Event**.

На рис.30 зображено частинки, які в результаті зіткнення з перегородкою стають джерелом других частинок і продовжують рухатися, а на рис.31 — частинки розкололися на зазначену кількість нових і після цього зникли.

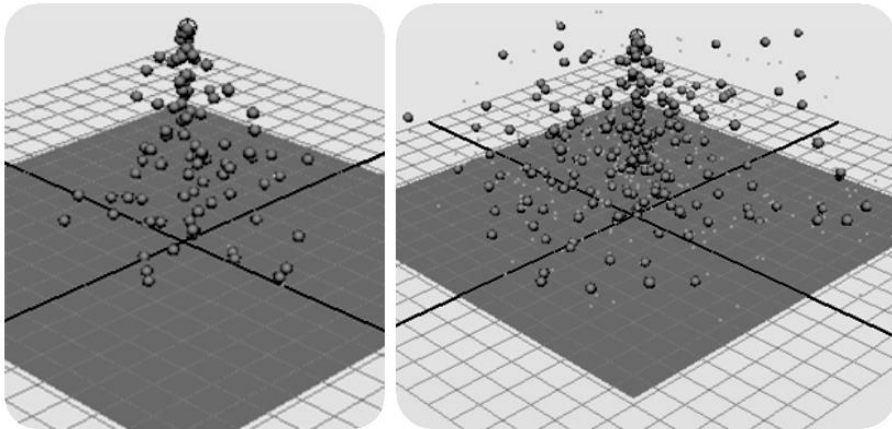


Рисунок 30 – Випускання нових частинок при зіткненні з перегородкою (подія типу **Emit**)

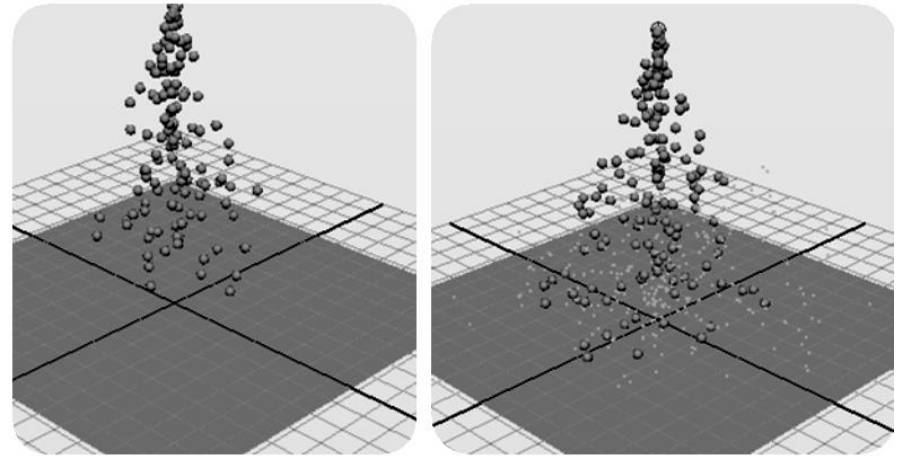


Рисунок 31 – Випускання нових частинок при зіткненні з перпоною (подія типу *Split*)

### ПРИТЯГУВАННЯ ЧАСТИНОК ДО МІШЕНІ

Потік частинок може притягуватися до мішені (цілі), найчастіше це використовується для того, щоб забезпечити повільний рух частинок, що буває важко реалізувати іншими методами. Мішенню може стати будь-який об'єкт, окрім кривої на поверхні. Якщо в якості мішені обирають полігональний об'єкт, то частинки будуть прямувати до його вершин, а якщо NURBS-поверхня, — то до керуючих точок. Коли мішенню буде інша система частинок, то перша створена в системі частинка, що притягується, буде прямувати за першою частинкою веденої системи, а остання — за останньою. Правда, даний принцип справедливий лише за умови, якщо частинки обох систем не помирають, бо в протилежному випадку передбачити порядок їх прямування неможливо.

За призначення мішені відповідає команда *Particles→Goal* (частинки→ціль). Сила тяжіння об'єкта мішенню залежить від індивідуальної ваги, яка змінюється в інтервалі від 0 до 1: при 0 мішень не впливає на об'єкт, а

при 1 її вплив буде максимальним (рис.32). Вага мішені визначається безпосередньо при її створенні або регулюється пізніше через редактор атрибутів. Конкретному об'єкту може бути призначено кілька притягуючих його мішеней — тоді об'єкт буде прямувати до кожної з мішеней одночасно, максимально наближуючись до тієї, яка має найбільшу вагу.

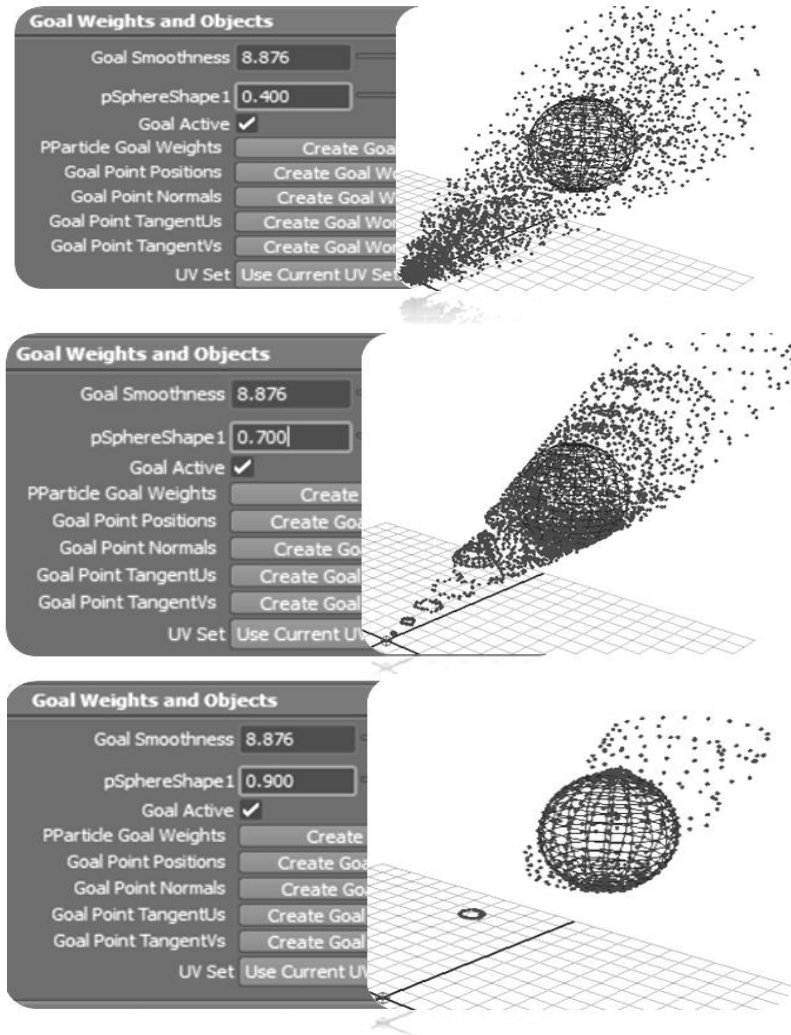
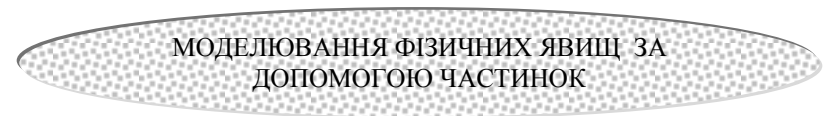


Рисунок 32 – Вплив індивідуальної ваги мішені на силу тяжіння частинок



Щоб спрямувати потік частинок в заданому напрямку до мішені треба спочатку виділити частинки, потім, при нажатій клавіші **Shift**, — об'єкт створений в якості мішені і спрямувати частинки до призначеної цілі командою **Particles** → **Goal**. Якщо виникає необхідність змінити силу тяжіння частинок до мішені, зменшуючи або збільшуючи цільову вагу останньої, — виділіть частинки, відкрийте редактор атрибутів, активуйте вкладку **particle Shape1** і розкрийте свиток **GoalWeight and Objects**. Змініть значення параметру ваги мішені. Коли вплив мішені треба ліквідувати (якщо вона стала не потрібною), можна зробити її неактивною, вимкнувши в свитку **GoalWeightandObjects** прапорець **Goal Active**, або взагалі видалити.

Скориставшись можливістю поставити ключі анімації, наприклад, на параметр **GoalWeight** (Цільова вага), можна отримати цікаві ефекти в діях частинок: різку зміну швидкості руху, кружляння навколо об'єкта або наліплювання об'єкта частинками, чи будь-що інше.



Вперше використовувати систему частинок для моделювання об'єктів з невизначеною формою, таких, наприклад, як хмари або вогонь запропонував V.T.Reev в 1983 році. Подальший розвиток комп'ютерної графіки розширив межі застосування частинок і урізноманітнив візуальні ефекти, які можна створювати за їх допомогою, впливаючи на форму, колір, прозорість, розмір, час існування, динаміку поведінки частинок та інші параметри. Розглянемо приклади моделювання деяких фізичних явищ, зокрема, диму та вогню двох типів: динамічних — з високою швидкістю розповсюдження та близьких до статичних — з повільним, м'яким характером поведінки.



1. Створіть джерело частинок **Omni** з наступними параметрами:
  - **Rate** =300;
  - **Speed** =0,02;

2. Додайте до частинок поле гравітації **Fields – gravity**, встановіть атрибуту **Magnitude** значення = -3 (рис.33).

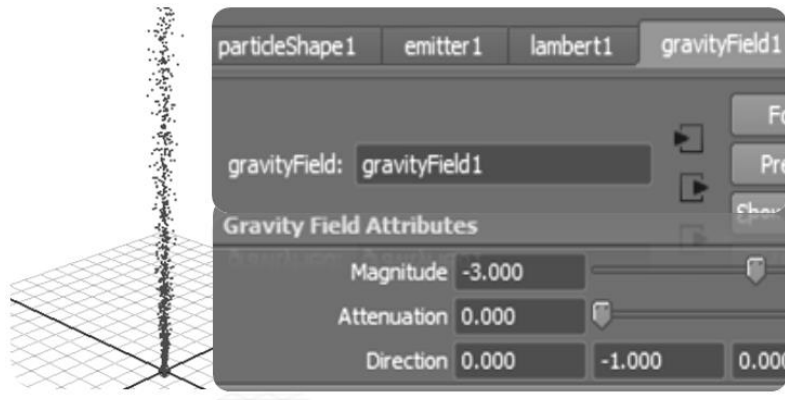


Рисунок 33 – Джерело частинок з заданими параметрами

3. Виділіть частинки і в **Attribute Editor** перейдіть в розділ **Lifespan Attributes**. Параметр **Lifespan Mode** встановлений в положення **Constant** задає для всіх частинок однаковий час існування, який дорівнюватиме одиниці. Варіація за час існування задається за допомогою параметра **Lifespan Mode=Random Range**. Задайте **Lifespan Random** =3, **Lifespan** =5 (рис.34).

4. Для того, щоб частинки рухаючись піддавались опору, використовуйте атрибут **conserve**, який імітує тертя з невидимим середовищем. В **Attribute Editor** в розділі **General Control Attributes** встановіть параметр **conserve** =0,95.



**ЗВЕРНИ УВАГУ!** Значення параметру **conserve**, як правило, задають близьким до одиниці, бо з математичної точки зору – це є коефіцієнт, на який помножують швидкість частинок в кожному кадрі, тому за секунду швидкість може знизитись дуже відчутно.

Виберіть частинки, створіть поле турбулентності **Fields –Turbulence** і встановіть атрибут **attenuation**=0 (рис.35). Для створення хаосу використо-

вують атрибут *Phase*, який задає зсув фази при обчисленні турбулентності. В *Attribute Editor* для поля турбулентності введіть в рамку атрибута *Phase X* наступний вираз *=noise(time)* (рис35).

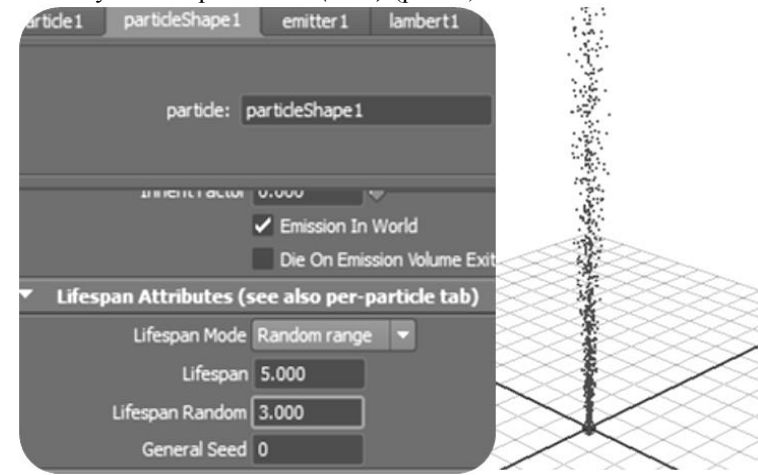


Рисунок 34 – Частинки з заданими варіаціями за час існування

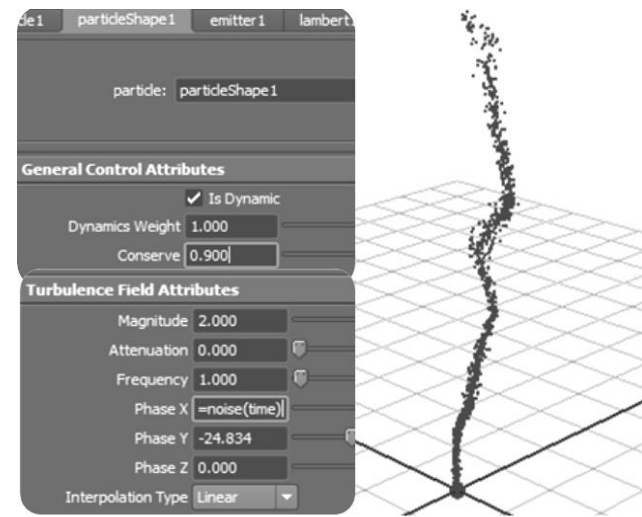


Рисунок 35– Додавання опору та хаосу до частинок

5. Якщо збільшити значення атрибута *Max Distance*, наприклад, до 0.658, то тоді частинки випускатимуться не із точки, а з невеликого сферичного об'єму (рис.36).

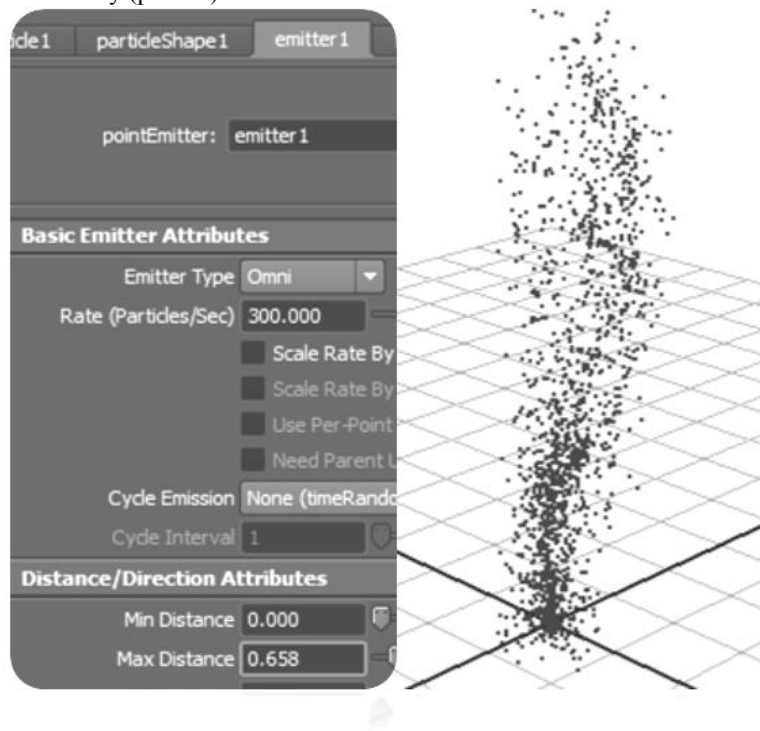


Рисунок 36– Збільшення об'єму джерела випускання частинок

6. Щоб додати диму прозорості виділіть частинки, відкрийте *Attribute Editor* і в розділі *Add Dynamic Attributes* натисніть кнопку *Opacity*. Із двох запропонованих варіантів вибираємо другий *Add Per Particle Attribute*. В розділі *Per Particle (Array) Attributes* відкриється атрибут *opacity PP*, на полі праворуч від імені атрибута натисніть праву кнопку і в меню, що з'явиться виберіть *Create Ramp* (рис.37). Для того, щоб відрегулювати ступінь прозорості натисніть праву кнопку на полі *opacity PP*. В контекстному меню виберіть *Edit Ramp*. Відрегулюйте текстуру так, щоб частинки спочатку мали деяку прозорість (сірий колір внизу), а починали зникати після половини прожитого часу (рис.38).

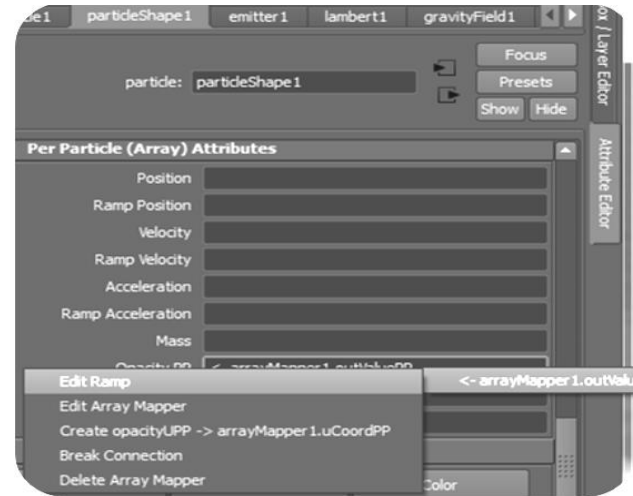
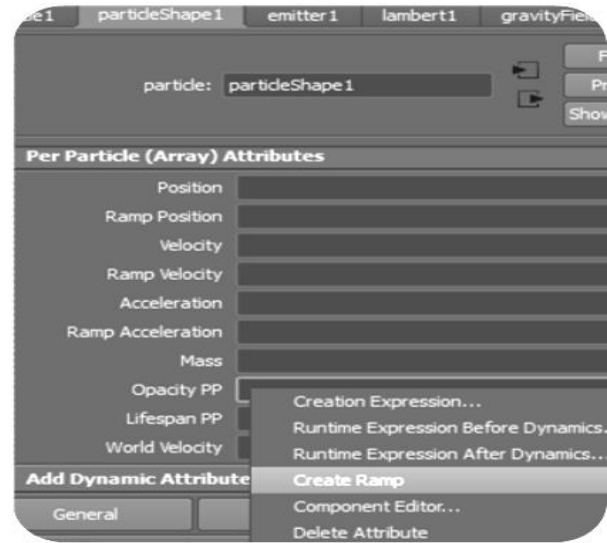


Рисунок 37 – Додавання прозорості частинкам за допомогою атрибуту *opacity PP* та створення текстури *Ramp*



Рисунок 38 – Налаштування текстури **Ramp** та вид частинок після візуалізації



### МОДЕЛЮВАННЯ ВОГНЮ

Розглянемо, як можна використати систему частинок для створення вогню, який би нагадував, наприклад, полум'я свічки.

1. Джерелом частинок буде **Directional**-емітер з потоком частинок спрямованим вздовж осі **Y**. Для емітера задаємо значення параметра **Spread** приблизно 0,2 (тоді частинки будуть випускатися в різних напрямках), швидкість частинок **Speed** задамо 2, і додамо їй випадковості **Speed Random** 1. (рис. 39).

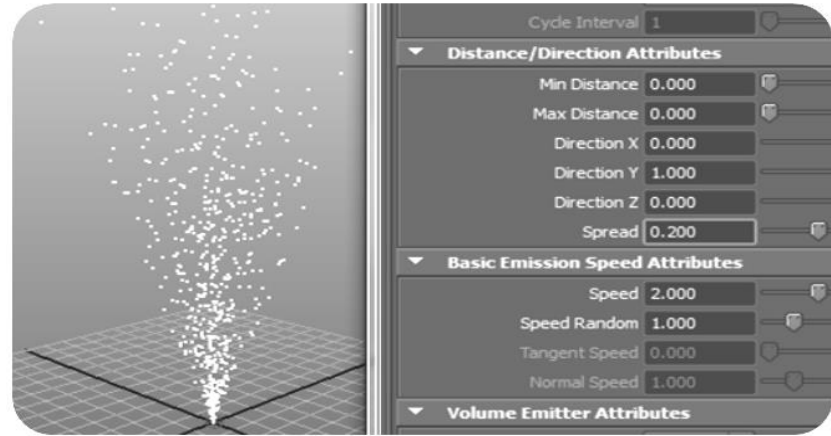
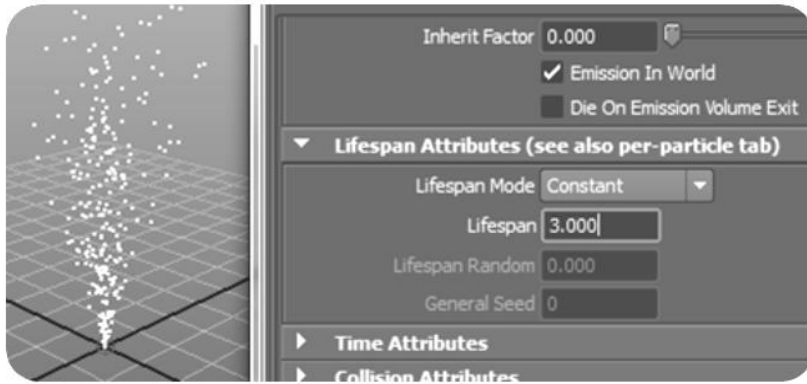
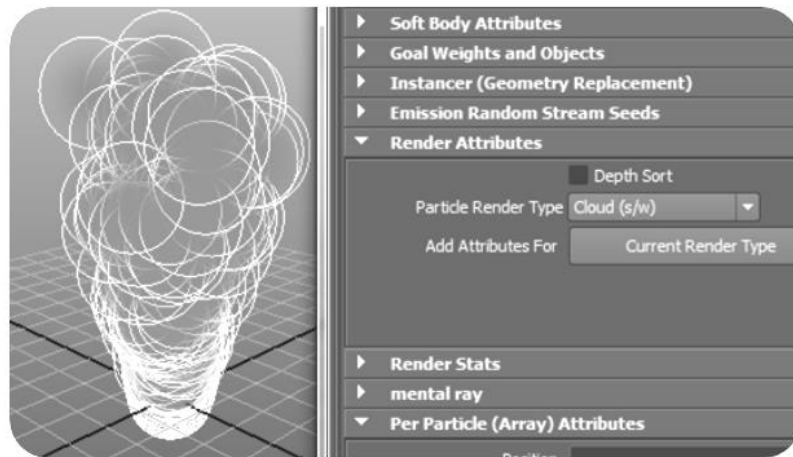


Рисунок 39 – Створення потоку частинок з заданими параметрами

2. В редакторі атрибутів на вкладниці **LifespanAttributes** (рис. 40,а) встановлюємо період існування частинок **Constant** і параметру **Lifespan** задаємо значення 3. Для візуалізації частинок вибираємо варіант **Cloud** (Хмара) — рис. 40,б.
3. Для того, щоб частинки мали неоднаковий розмір, при виділеній системі частинок в редакторі атрибутів в розділі **AddDynamicAttributes** (Додавання динамічних атрибутів) треба натиснути на кнопку **General** (Загальні), відкрити вкладку **Particle** (Частинки), виділити рядок **radiusPP** і натиснути на кнопку **Ok** (рис.41,а). Результатом цих дій буде поява в розділі **PerParticleArrayAttributes** (Атрибути однієї частинки масива) нового атрибута **radiusPP** (Радіус однієї частинки). Натиск правої кнопки на полі **radiusPP**, яке з'явилося в розділі **PerParticleArrayAttributes**, дає можливість в контекстному меню вибрати команду **CreateRamp** (Створити градієнт) (рис.41,б).
4. Ще один натиск на полі **radiusPP** правою кнопкою миші, дозволяє вибрати із контекстного меню команду **Edit Ramp** (Редагувати градієнт). Якщо параметри градієнта відрегулювати як показано на рис.42 а, то радіус у **Cloud**-частинок зменшуватиметься із зростанням часу їх існування (рис.42 б).



a



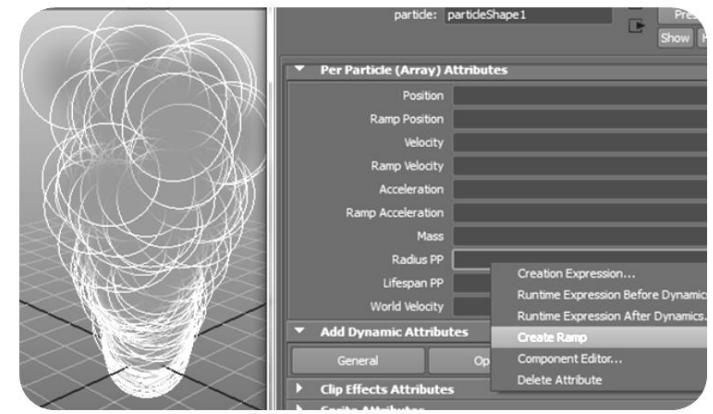
б

Рисунок 40 – Обмеження часу існування частинок *Lifespan Constant* (а) та вид частинок після вибору візуалізатора *Cloud* (б)



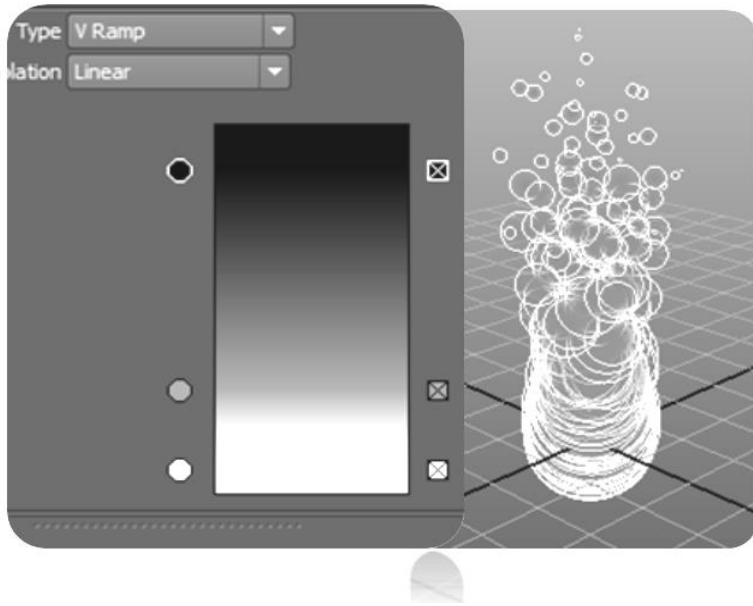


а



б

Рисунок 41 – Додавання динамічного радіуса (а) до атрибутів частинок та вибір текстурного градієнта **RAMP** (б) для редагування розміру частинок



а

б

Рисунок 42 – Налаштування поля градієнта **RAMP** для впливу на розмір частинок (а) і вид частинок після додавання динамічного радіуса (б)

5. Можна додати потоку частинок ефект коливання, відхиляючи полум'я то в одну, то в другу сторону. Для цього треба створити мішені для системи частинок і розмістити їх по різні боки від полум'я. В якості мішеней можуть бути застосовані, наприклад, сфери (рис. 43). Спочатку виділяються частинки, потім при нажатій клавіші **Shift** — першу сферу, далі за допомогою команди **Particles=>Goal** частинки націлюють на сферу. Такі ж дії повторюють по відношенню до другої сфери.
6. Цільова вага обох мішеней однакова — як між собою, так і протягом створеної анімації. Для кожної мишені треба підібрати вагу, враховуючи при цьому номер кадра. В прикладі, що розглядається, довжина анімації складає 70 кадрів. В 1-му кадрі перша сфера має вагу 0,150, друга — 0, на обидві ваги встановлено ключі (рис. 44). В 21-му кадрі вага сфер дорівнює 0,100 і 0,150 відповідно (рис. 45), в 41-му — 0,150 і 0,100, а в 61-му — 0,100 і 0,150. Після цього потік частинок відхилятиметься бажаним чином (рис. 46).

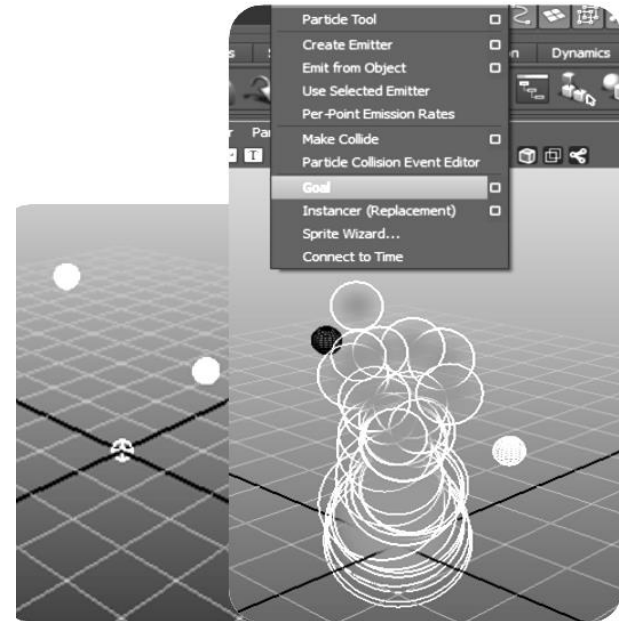


Рисунок 43 – Додавання двох сфер в сцену

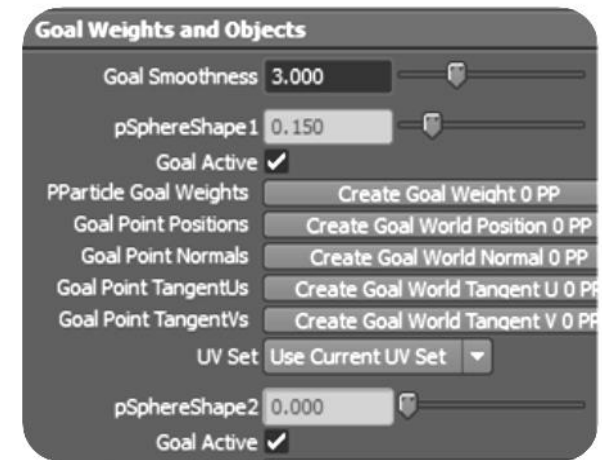


Рисунок 44 –Створення ключів для цільової ваги в 1-му кадрі

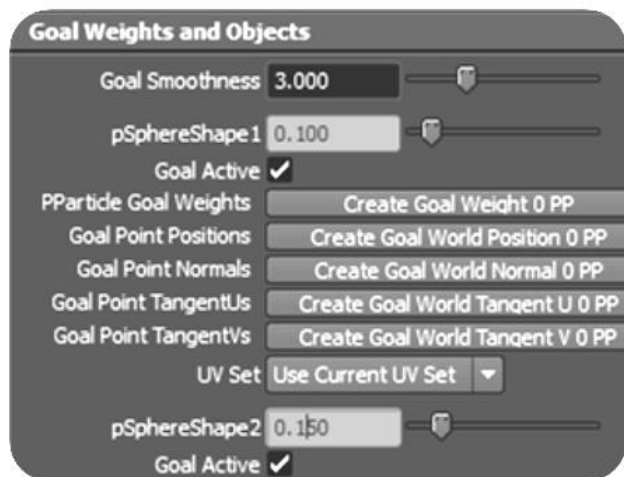


Рисунок 45 –Створення ключів для цільової ваги в 21-му кадрі

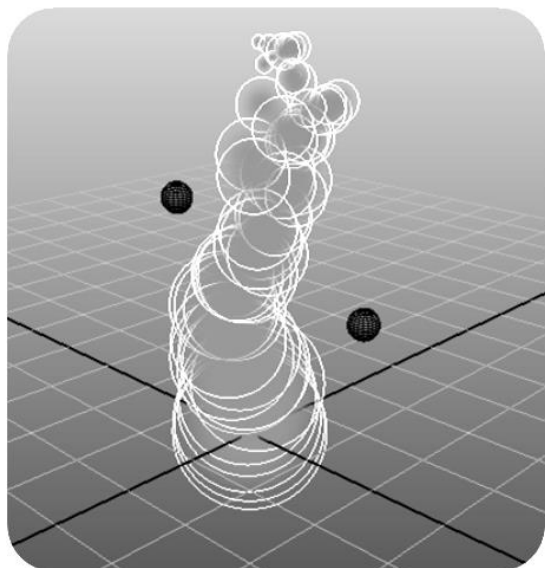


Рисунок 46 – Вид потоку частинок в одному із проміжних кадрів

7. Наступним кроком буде створення матеріалу для полум'я. Але спочатку додаємо до частинок атрибут прозорості і зробимо їх біля джерела горіння практично прозорими, з подальшим зменшенням прозорості. Виділяємо систему частинок і в закладці **Add Dynamic Attributes** редактора атрибутів натискаємо на кнопку **Opacity** (Непрозорість). В вікні, що відкривається, включаємо галку для **Add Per Particle Attribute** (Додати атрибут для частинок) — рис. 47 і натискаємо на кнопку **Add Attribute** (Додати атрибут). З'явиться новий атрибут **opacityPP** на вкладці **PerParticleArrayAttributes**. Натискаємо праву кнопку на полі **opacityPP** і вибираємо в контекстному меню команду **Create Ramp** (Створити градієнт). Клацнемо на полі **opacityPP** правою кнопкою миші, вибираємо із контекстного меню команду **Edit Ramp** (Редагувати градієнт) і змінюємо параметри градієнта відповідно до рис. 48. Після цього прозорість частинок стане залежати від їхнього віку.



Рисунок 47 –Додавання атрибута прозорості

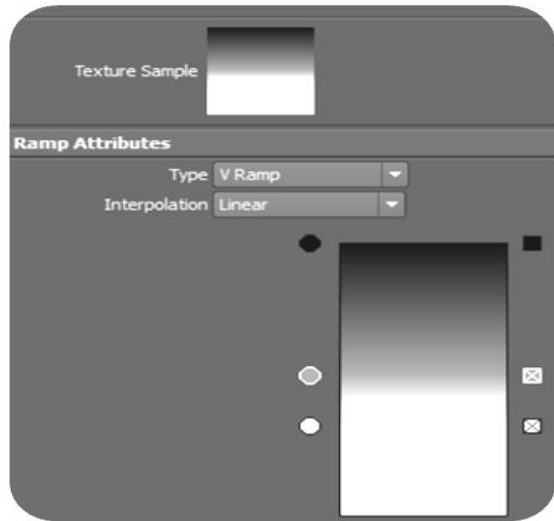


Рисунок 48 – Параметри настройки градиента для динамичної прозорості

Далі редагуємо автоматично призначений системі частинок об'ємний матеріал *particleCloud1*, який знаходиться в редакторі матеріалів. На рис. 49 наведено зображення одного із проміжних кадрів створеної анімації.

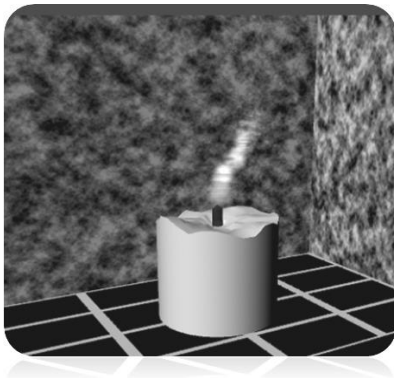


Рисунок 49 – Полум'я свічки

## МОДЕЛЮВАННЯ СНІГУ

Розглянемо приклад створення снігу за допомогою частинок. В якості джерела частинок використаємо полігональну площину. В режимі *Dynamics* задаємо *Particles* → *Emit from Object* (рис.50) з параметрами:

*emitter Type* – *Directional* ; *Speed* – 1; *Speed Random* – 2; *Spread* – 0,2.

Виділяємо частинки і додаємо поле гравітації *Gravity* з амплітудою *Magnitude* – 2 і напрямком *-Y* (рис.51) та поле турбулентності *Turbulence* з амплітудою *Magnitude* – 5, затуханням *Attenuation* – 0 (рис.52). При програнні сцени частинки рухаються подібно сніжинкам, що падають та кружляють з невеликою хаотичністю. Скориставшись можливістю замість частинок підставляти будь-який об'єкт за допомогою команди *Instancer*, замінюємо наші частинки змодельованою сніжинкою (рис.53). І нарешті, задаємо період існування частинок в сцені. В розділі *Lifespan Attributes* встановлюємо *Lifespan mode* – *Constant*, і задаємо значення *Lifespan* – 5. Частинки (сніжинки) існують в сцені 5 секунд, після чого зникають (рис.54).

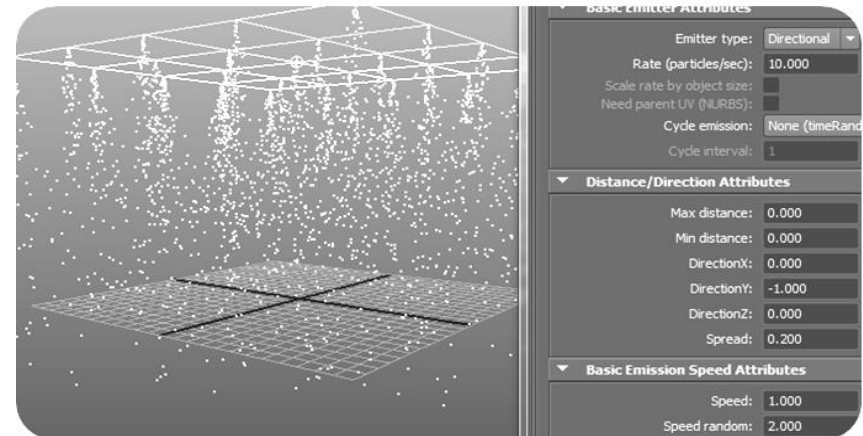


Рисунок 50 – Основні параметри створеного емітера *Emit from Object*

Якщо поставити ключі анімації на параметр **Rotate** основної сніжинки (**Instancer**), і заставити її, наприклад, кружляти, то тоді всі сніжинки теж будуть повторювати ці рухи.

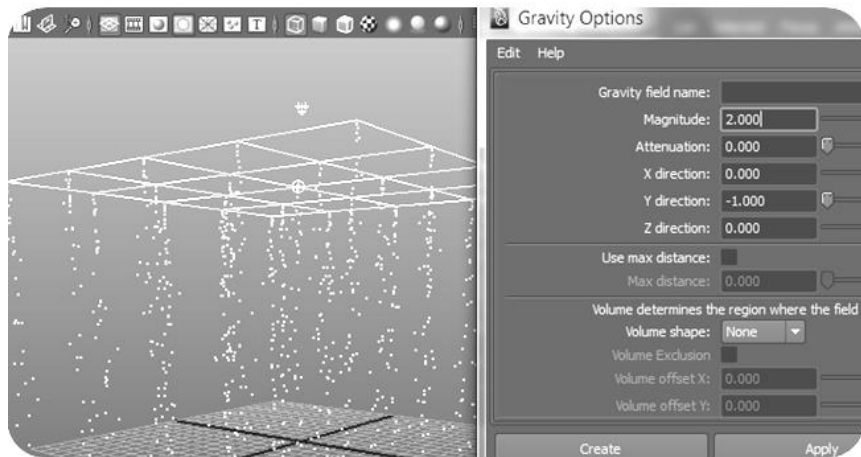


Рисунок 51 – Основні параметри доданого до частинок поля гравітації

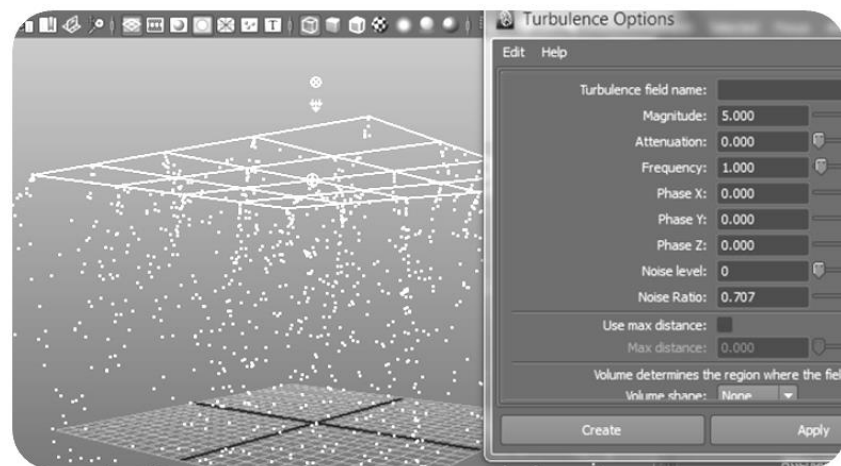


Рисунок 52 – Основні параметри доданого до частинок поля турбулентності



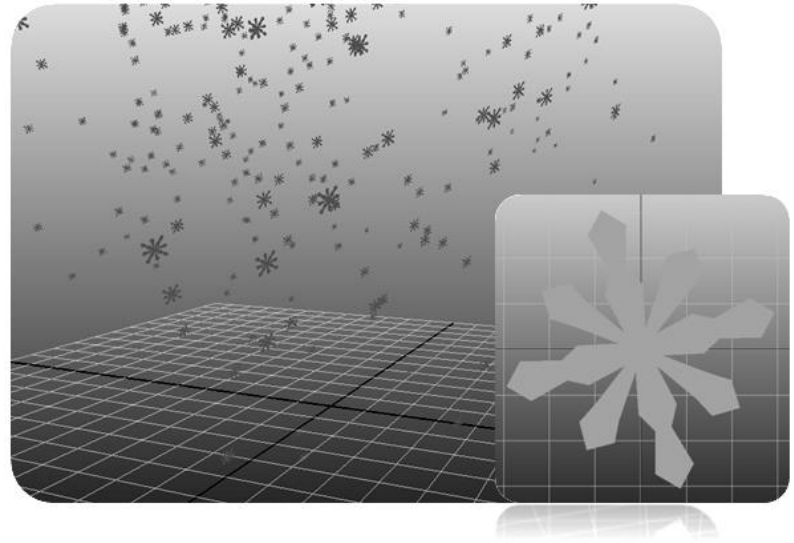


Рисунок 53 – Підстановка (*Instancer*) сніжинок замість частинок в сцену

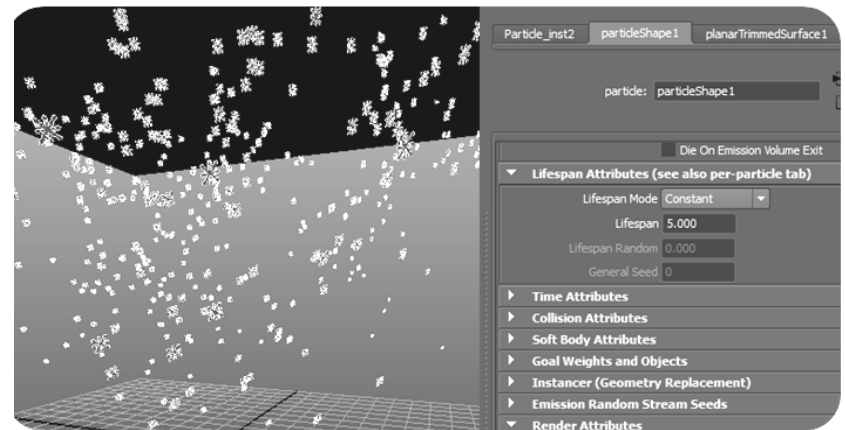


Рисунок 54 – Обмеження часу існування частинок в сцені

На рис.55 один із кадрів створеної анімації, де додано фоновий рисунок з зимовим пейзажем.



Рисунок 55 – Додавання фонового рисунка в сцену

## КОНТРОЛЬНІ ЗАПИТАННЯ

1. Назвіть способи створення частинок.
2. Які основні атрибути має джерело частинок?
3. Які атрибути частинок ви знаєте?
4. В якій послідовності додають силові поля до частинок? З якою метою це роблять?
5. Назвіть основні типи силових полів та їхні основні атрибути.
6. Для чого використовують редактор динамічних зв'язків?
7. Як впливає індивідуальна вага мішені на силу притягування частинок?
8. Яким чином можна задавати колір кожній частинці?

## СПИСОК ЛІТЕРАТУРИ

3. Риддел Денни, Даймонд Адриан. Maya для Windows и Macintosh/ Риддел Денни, Даймонд Адриан М: 2007.
4. Кундерт-Гиббс Дж., Ларкинс М., Деракшани Д. Освоение Maya 8.5/ Москва: 2007.
5. Марк Р. Улкинс, Крис Казмиер. Maya: программирование на MEL.
6. Цыпцын С., Забелин В., Король Г., Ледин П. Мастер класс по Autodesk Maya, 2006.
7. Цыпцын С. Понимая Maya. Книга первая и вторая 2006//Athouse media, 2007. – 1428 с.
8. Риддел Д. Maya 5 для Windows и Macintosh, 2004.
9. Дэвид А. Д. Гоулд, Полное руководство по программированию Maya, 2004.
10. М. Адамс, Э. Миллер, М. Симс, Maya 5. Для профессионалов, 2004.

## ЗМІСТ

Вступ.....	3
Природа частинок.....	4
Створення частинок.....	4
Атрибути генератора частинок.....	7
Атрибути частинок.....	8
Силкові поля.....	22
Загальні атрибути полів.....	23
Зіткнення частинок з перепонами.....	24
Притягування частинок до мішені .....	30
Моделювання фізичних явищ за допомогою частинок.....	32
Моделювання диму.....	32
Моделювання вогню.....	37
Моделювання снігу.....	46
Контрольні запитання.....	50
Список літератури.....	51

Навчальне видання

**МОДЕЛЮВАННЯ ФІЗИЧНИХ ЯВИЩ  
ІЗ ЗАСТОСУВАННЯМ ДИНАМІКИ ЧАСТИНОК**

*Методичні вказівки  
для лабораторних занять  
та самостійної роботи студентів із дисципліни  
«Моделювання спеціальних ефектів в комп'ютерній графіці»*

Укладачі: САВЧЕНКО Лідія Митрофанівна  
ВОРОНЦОВА Дар'я Володимирівна  
РОЖЕНКО Зінаїда Михайлівна

Відповідальний за випуск О.В.Шоман

Роботу рекомендував до видання А. М. Краснокутський

В авторській редакції

План 2012 р., поз. 42

Підп. до друку                      Формат 60x84 1/16. Папір офісний.  
RISO- друк. Гарнітура Таймс. Ум. друк. арк.      Обл.-вид. арк. 1,4.  
Наклад 50 прим. Зам. №              Ціна договірна.

---

Видавничий центр НТУ «ХПІ».  
Свідоцтво про державну реєстрацію ДК № 116 від 10.07. 2000 р.  
61002, Харків, вул. Фрунзе, 21

---

Друкарня НТУ «ХПІ», 61002, Харків, вул. Фрунзе, 21

---