

МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ
КРИВОРІЗЬКИЙ ДЕРЖАВНИЙ ПЕДАГОГІЧНИЙ УНІВЕРСИТЕТ»
Фізико-математичний факультет
Кафедра інформатики та прикладної математики

«Допущено до захисту»

Завідувач кафедри

Реєстраційний № _____

_____ Соловйов В.М.

«___» _____ 2019 р.

«___» _____ 2019 р.

СТВОРЕННЯ 3D-АНІМАЦІЙНИХ ПРЕЗЕНТАЦІЙ
З ВИКОРИСТАННЯМ HOUDINI

Кваліфікаційна робота студента
фізико-математичного факультету
групи Ім-14
ступінь вищої освіти «магістр»
спеціальності
014.09 Середня освіта (Інформатика)
Голуба Олександра Івановича

Керівник к.ф.-м.н., доцент
Моїсеєнко Н.В.

Оцінка:

Національна шкала _____

Шкала ECTS __ Кількість балів _____

Голова ЕК: _____

Члени ЕК: _____

ЗМІСТ

ВСТУП	3
РОЗДІЛ 1 КОМП'ЮТЕРНА ГРАФІКА ТА СФЕРИ ЇЇ ЗАСТОСУВАННЯ.....	5
1.1. Сфери використання комп'ютерної графіки	5
1.2. Технології створення 3D графіки	13
1.3. Програмне забезпечення для створення тривимірної графіки	21
Висновки до розділу 1	31
РОЗДІЛ 2 ФІЗИЧНЕ МОДЕЛЮВАННЯ В КОМП'ЮТЕРНІЙ ГРАФІЦІ ТА ЙОГО РЕАЛІЗАЦІЯ В ПАКЕТІ HOUDINI.....	32
2.1. Пакет процедурного моделювання Houdini	32
2.2. Фізичне моделювання в комп'ютерній графіці	44
Висновки до розділу 2	52
РОЗДІЛ 3 ПРОГРАМНА РЕАЛІЗАЦІЯ.....	53
3.1. Створення 3D симуляції засобами Houdini	53
3.2. Тестування та аналіз проблем.....	60
Висновки до розділу 3	65
ВИСНОВКИ.....	65
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	67

ВСТУП

Актуальність теми. Прискорення темпів життя і безпосередньо пов'язане з ним зростання обсягу інформаційного потоку породжує проблематику відбору інформації, виділення головного і фільтрації зайвого. Основний спосіб сприйняття інформації залишається строго візуальним з певним зміщенням від тексту до зображення. Візуальне сприйняття відіграє велику роль у нашому повсякденному житті. Воно допомагає нам у взаємодії з людьми та у навчанні. Інформація у вигляді зображень засвоюється швидше, ніж у текстовому вигляді та краще запам'ятовується.

Поява і розвиток засобів комп'ютерної графіки відкриває для сфери освіти принципово нові потенціали, завдяки яким з'явилася можливість не тільки використовувати графічні образи в якості ілюстрацій, а й змінювати їх на свій розсуд, дослідити поведінку об'єктів, динамічно управляти їх змістом, формою, розмірами і кольором, досягаючи найбільшої наочності.

Сьогодні комп'ютерна графіка - це один з напрямків інформаційних технологій, який найбільш бурхливо розвивається. За допомогою комп'ютерної графіки можна зробити зримими або візуалізувати такі явища і процеси, які не можуть бути побачені в дійсності, можна створити наочний образ того, що насправді ніякої наочності не має (наприклад, ефекти теорії відносності, закономірності числових рядів тощо).

Houdini - це один з найпотужніших програмних додатків, який містить процедурні інструменти для створення візуального контенту для візуалізацій, кіно, телебачення та ігор.

Об'єкт дослідження. Комп'ютерна графіка, та методи її створення

Предмет дослідження. Використання рушія Houdini для розробки 3D візуалізацій.

Мета. Створити 3D анімаційну презентацію за допомогою рушія Houdini.

Для досягнення мети необхідно розв'язати такі завдання:

- проаналізувати підходи до створення 3D графіки;
- порівняти наявні програмні засоби для роботи з 3D графікою, виявити їх переваги та недоліки;
- обрати інструментарій для роботи;
- проаналізувати принципи фізичного моделювання;
- реалізувати за допомогою рушія Houdini симуляцію потоку рідин.

Структура роботи. Кваліфікаційна робота складається з 3 розділів та висновків, містить 49 рисунків та 25 використаних джерел.

РОЗДІЛ 1

КОМП'ЮТЕРНА ГРАФІКА ТА СФЕРИ ЇЇ ЗАСТОСУВАННЯ

1.1. Сфери використання комп'ютерної графіки

Комп'ютерна графіка – це область комп'ютерних наук, яка вивчає методи цифрового синтезу і маніпулювання візуальним контентом. Хоча цей термін часто відноситься до вивчення тривимірної комп'ютерної графіки, він також охоплює двовимірну графіку і обробку зображень [3]. Графіка часто відрізняється від області візуалізації, хоча у цих двох понять багато спільного. Розваги (у вигляді анімаційних фільмів і відеоігор) є, мабуть, найпопулярнішою областю комп'ютерної графіки [9].

Двовимірну графіку зазвичай поділяють на векторну і растрову.

На сторінках web-сайтів векторну графіку можна побачити в якості шрифтів, іконок, логотипів. Вектор може без втрат масштабуватися, деформуватися, також імітувати тривимірну графіку, чим не може похвалитися растрова графіка, яка бере за основу пікселі (рис. 1.1.). Як приклад растрової графіки - будь-які картинки, зображення, фотографії, які складаються з матриці пікселів.

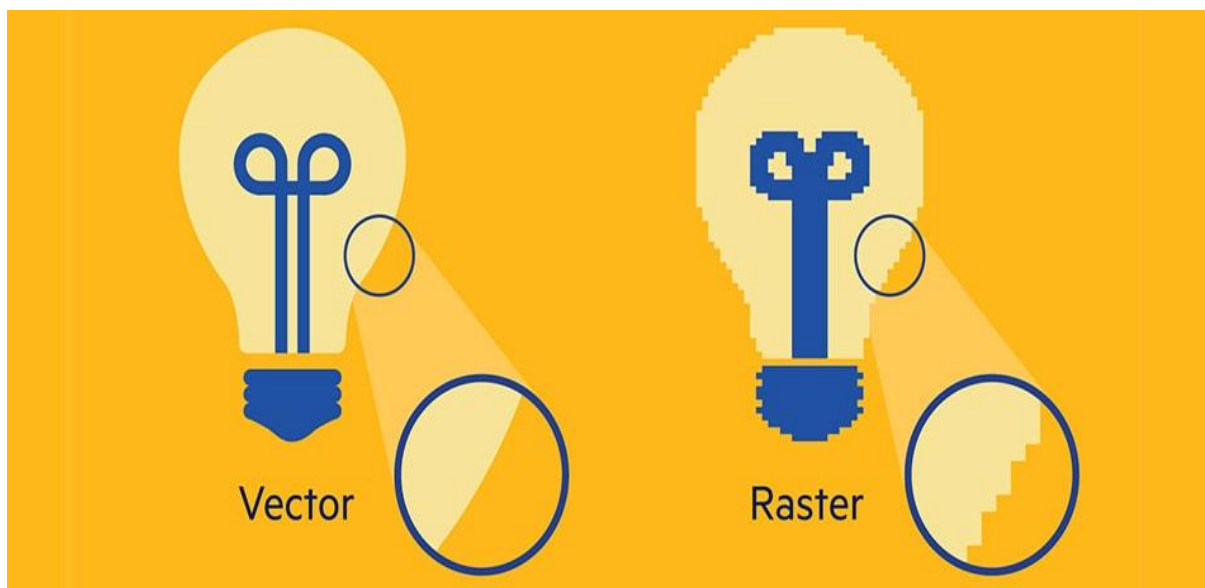


Рис. 1.1. Растрова та векторна графіка

3D-графіка або тривимірна графіка - це сфера комп'ютерної графіки, набір методів та інструментів, що дозволяють створювати тривимірні об'єкти з використанням форм і кольорів. Вона відрізняється від двовимірних зображень створенням геометричній проекції тривимірної моделі сцени (віртуального простору) в 2D [1]. Це реалізується за допомогою спеціалізованого програмного забезпечення. Отримана модель може бути копією об'єкта реального світу (наприклад, будівля, людина, автомобіль, астероїд), або бути повністю абстрактною. Тривимірна модель - це математичне уявлення будь-якого тривимірного об'єкту (неживого або живого). Однією з найвідоміших моделей являється "чайник з Юти" - тривимірна модель реального чайника, яку часто застосовують для тестування або демонстрації в комп'ютерній графіці. Примітив чайника вважається еквівалентом програми «Hello, World» як спосіб створення простої тривимірної сцени з дещо складною моделлю, яка виступає у якості базового геометричного орієнтира для налаштування сцени і освітлення (рис. 1.2.).

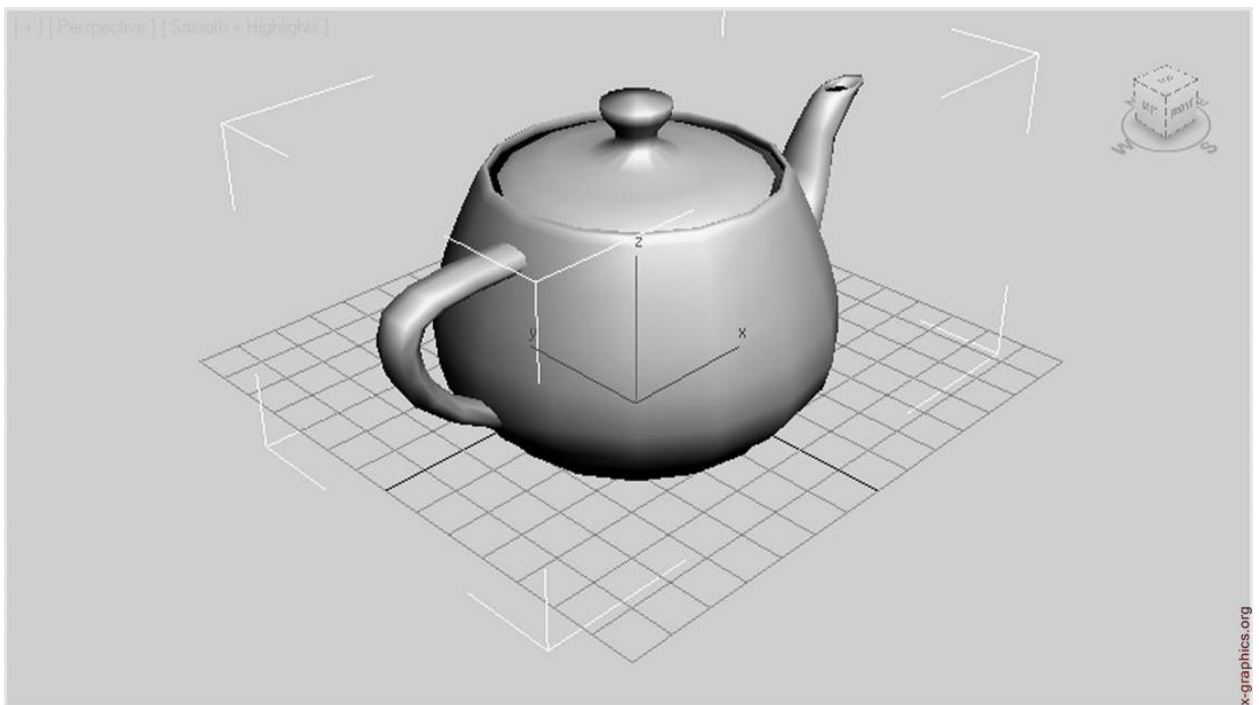


Рис. 1.2. 3D модель чайника з Юти

Модель технічно не є графікою, поки вона не буде візуально відображена. Завдяки 3D-друку 3D-моделі не обмежуються віртуальним простором.

Модель може відображатися візуально як двовимірне зображення за допомогою процесу, званого 3D-рендерингом, або використовуватися в неграфічних комп'ютерних симуляції і обчисленнях.

Тривимірна графіка та моделювання мають багато напрямків використання. Виробництво та реклама: 3D моделювання часто застосовується при створенні нових видів продукції, або коли наявне двовимірне зображення предмета є недостатнім для реалізації поставлених цілей [18].

На основі 3d-моделі, за допомогою технологій швидкого виготовлення прототипу (3d друк, фрезерування, лиття силіконових форм і т.д.), створюється реалістична модель майбутнього виробу.

3D-моделі і їх візуалізація часто застосовуються в презентаціях, Інтернет-рекламі, телерекламі та інших видах комерційної діяльності (рис. 1.3.).



Рис. 1.3. 3D дизайн виробів

Медицина: 3D-технології зробили справжню революцію в медицині. За допомогою 3D-моделювання, 3D-сканування і 3D-друку стало можливим створювати медичні вироби, максимально адаптовані під індивідуальні особливості людини. 3D-сканери та 3D-принтери в хірургії та стоматології сьогодні - звичайна практика.

Нові технології дають можливість виготовляти макети органів і кісток, створювати 3D-моделі і виконувати високоточний друк на 3D-принтері протезів, імплантів, ортопедичних устілок та інших виробів (рис. 1.4.).

Завдяки цим інноваціям вдається підвищити надійність продукції, заощадити час, знизити виробничі витрати і вартість кінцевих виробів, а головне - поліпшити життя пацієнтів.

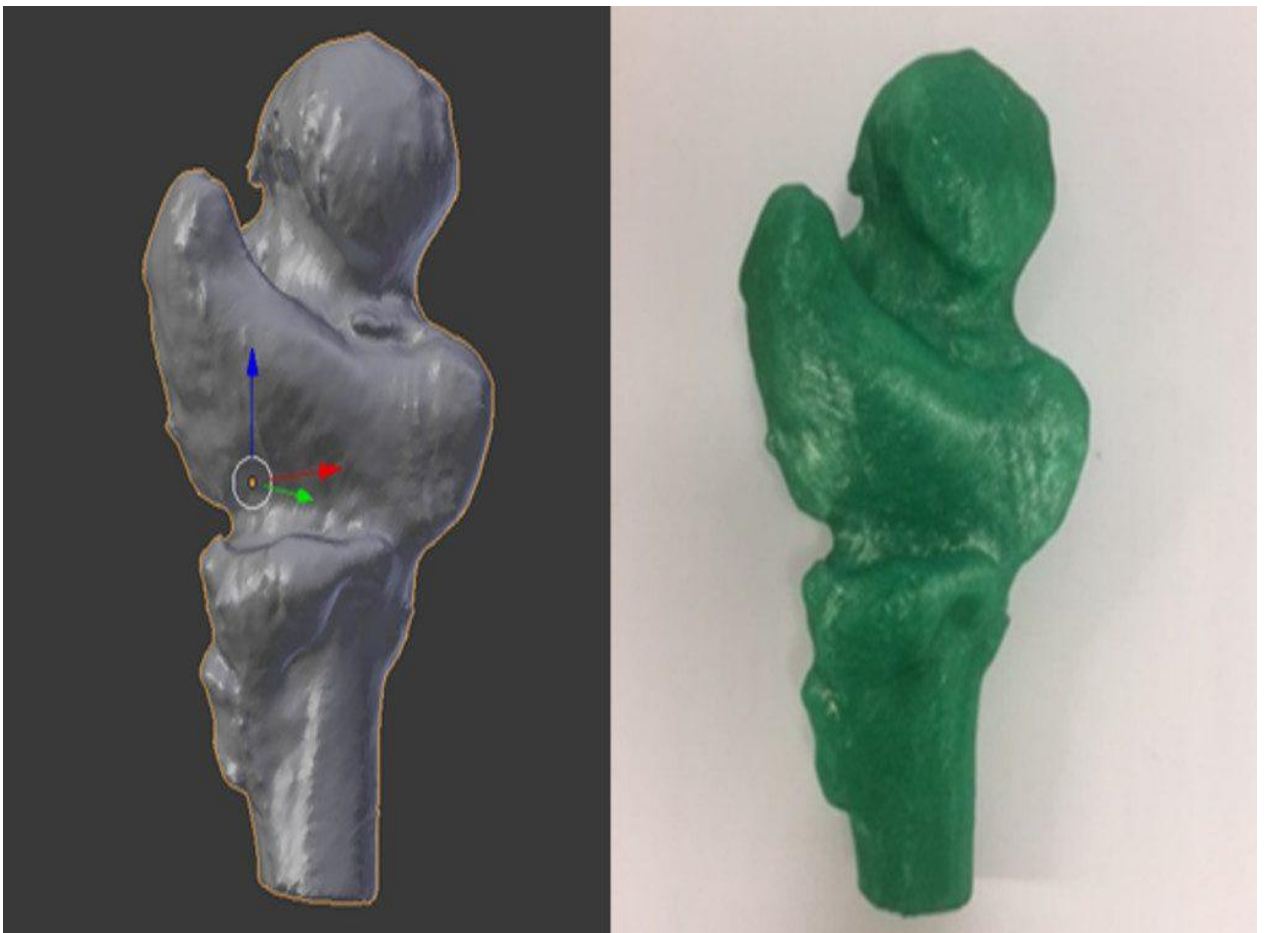


Рис. 1.4. 3D модель кістки

Архітектурна візуалізація: Тривимірні технології сьогодні вирішують різноманітні завдання в будівництві та архітектурі, дозволяючи прискорити і спростити процес проектування.

3D-модельовання будівель, споруд, приміщень, пам'ятників та інших об'єктів допомагає швидше донести ідеї до замовника і вносити корективи на будь-якій стадії проекту (рис. 1.5.).

Якщо перед вами стоїть завдання реконструювати або модернізувати об'єкт, розробити технічну документацію або з високою точністю виміряти геометрію будівлі, отримати 3D-модель можна за допомогою промислового 3D-сканера [1].

Архітектурна візуалізація допомагає архітекторам і дизайнерам працювати спільно і більш ефективно обмінюватися ідеями.

Візуалізація архітектури дозволяє побачити різницю між теоретичними принципами концептів і їх практичним застосуванням.



Рис. 1.5. Архітектурна візуалізація будинку

3D-анімація - творчий процес, в ході якого ведеться маніпулювання тривимірними моделями або об'єктами для експорту послідовностей зображень, створюючи ілюзію анімації або руху. Цей же принцип використовується в 2D анімації.

Єдина різниця між 2D і 3D-анімацією полягає в тому, що в 2D зображення малюються від руки, а зображення 3D анімації генеруються комп'ютером.

Процедура створення 3D послідовно поділяється на три основні розділи: моделювання, верстка, анімація і рендеринг.

Сьогодні велика частина 3D-мультфільмів робиться за допомогою комп'ютерної графіки або CGI (computer-generated imagery) (рис. 1.6.). Найвідомішою студією CGI-анімації є Ріхар [1]. Саме ця компанія, заснована в 1985 році, створила перший у світі CGI фільм.



Рис. 1.6. 3D-анімаційні мультфільми

Майже кожна велика комп'ютерна гра, випущена в наші дні, зроблена в 3D або використовує велику кількість 3D-ресурсів (рис. 1.7.). Ігри - це унікальне середовище для роботи, як в цілому, так і як у вигляді форми мистецтва, тому що, на відміну від майже будь-якої іншої форми вираження, ігри є інтерактивними.

Через це ігри запускаються на ігрових рушіях, які приймають на себе дії гравця і виводять результати, зокрема, вони використовують рендеринг в реальному часі для створення тривимірної графіки, яку ви бачите під час гри.

Рендеринг у реальному часі - це коли комп'ютерна система генерує зображення або результати дій для глядача / гравця під час їх виникнення.



Рис. 1.7. 3D графіка у сучасній відео-грі

Наукова візуалізація та комп'ютерне моделювання: Наукова візуалізація, яку іноді скорочено називають SciVis, - це представлення даних графічно як засобу для вивчення та розуміння даних. Іноді такий метод називають візуальним аналізом даних. Це дозволяє досліднику отримати

уявлення про систему, яка вивчається раніше неможливими способами. Важливо розрізняти наукову візуалізацію та графічну презентацію. Графічна презентація створюється насамперед для передачі інформації та результатів способами, які легко зрозуміти [16]. У науковій візуалізації переважає прагнення зрозуміти дані. Однак часто обидва способи переплітаються (рис. 1.8.).

З точки зору обчислень, SciVis є частиною більшого поля, яке називається візуалізацією. Воно включає дослідження в комп'ютерній графіці, обробці зображень, високоефективних обчисленнях та інших областях. Ті ж інструменти, які використовуються для SciVis, можуть застосовуватися, наприклад, для анімації або мультимедійної презентації.

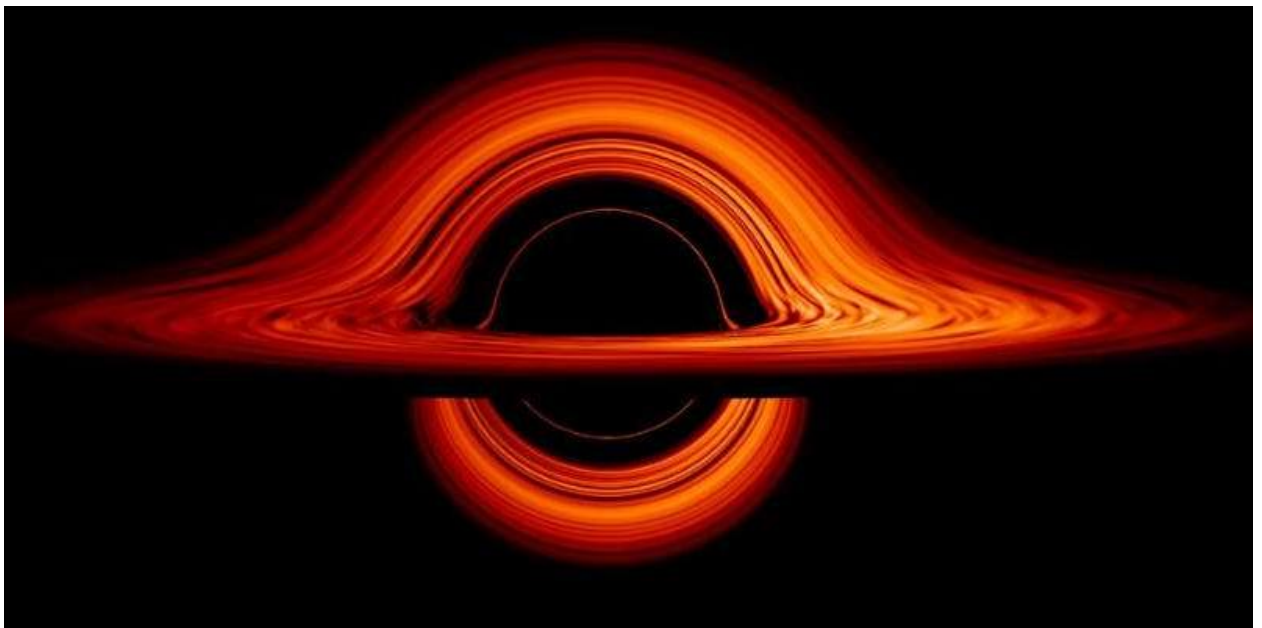


Рис. 1.8. Графічна візуалізація чорної діри

1.2. Технології створення 3D графіки

Полігональне моделювання

Найпопулярніша технологія створення об'єктів в тривимірній графіці - створення полігональної сітки. Це означає, що об'єкт описується вершинами, з'єднуючими їх ребрами та гранями.

Тобто будь-який персонаж гри або кіно - геометрична фігура, що складається з множини граней [18]. Ви створюєте сітку і деформуєте, поки вона не приймає форму потрібного об'єкта.

Всі 3D-сітки будуються з трьох основних елементів: Vertices (Вершини), Edges (Ребра), Polygons (Полігони) (рис. 1.3.).

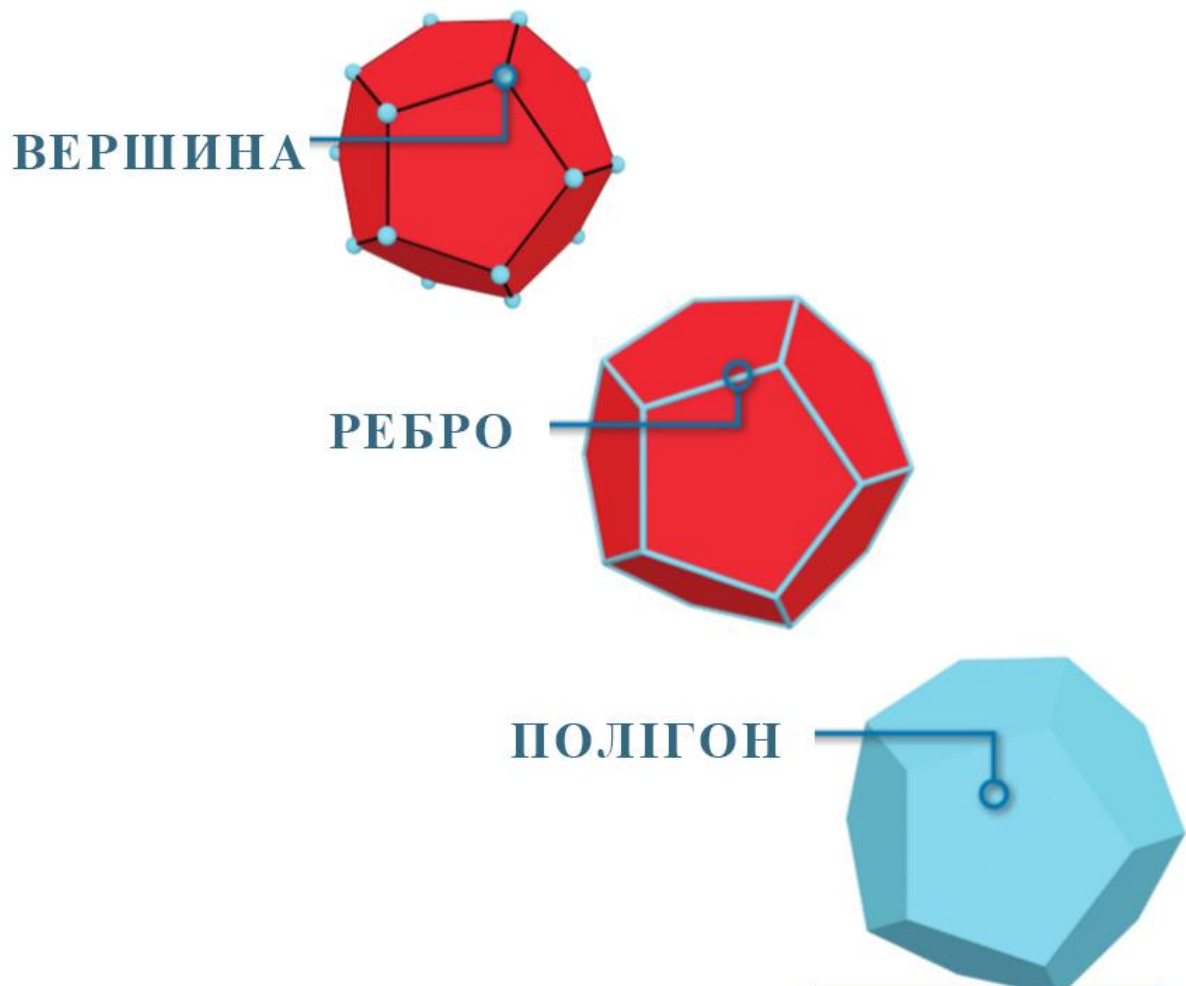


Рис. 1.3. Базові елементи 3D-сітки

- Vertex (Вершина) (множина: vertices або vertexes) є елементом найнижчого рівня, що становить 3D-сітку. По суті, vertex - це просто позиція точки в тривимірному просторі, яка визначається її координатами X, Y і Z. Кілька ребер і полігонів можуть мати спільні вершини.
- Edge (Ребро) (множина: edges) - це одновимірна лінія, яка з'єднує дві вершини. Як і у вершин, кілька полігонів можуть мати спільні ребра.
- Polygon (Полігон, тобто багатокутник) (множина: polygons) визначається трьома або більшою кількістю вершин і має замкнутий контур з ребер.

Ці геометричні форми визначають поверхню тривимірної сітки. Типова тривимірна сітка складається з сотень, тисяч або навіть мільйонів полігонів. Полігони також іноді називають poly або face (гранню, поверхнею).

На початковому етапі конструювання, створюються нізкополігональні моделі, що дає можливість скоротити час на обробку даних. При цьому рівень деталізації буде невисоким. Щоб створити модель з більш проробленою деталізацією, необхідно збільшити число полігонів, така модель буде називатися високополігональною (рис. 1.4.).

Метод застосовується, коли необхідно створити точну копію об'єкту. Чим більше полігонів на площу моделі, тим точніше модель.

Однак, це не означає, що якщо модель містить мало полігонів (low poly), то це погана модель. Теж саме, можна сказати про те, що якщо в моделі кілька мільйонів полігонів (High poly), то це добре.

Все залежить від призначення. Наприклад для ігор використовують моделі, які складаються не більше як з декількох десятків тисяч полігонів. Для кіно або анімації використовують моделі, кількість полігонів яких може досягати до декількох десятків мільйонів.

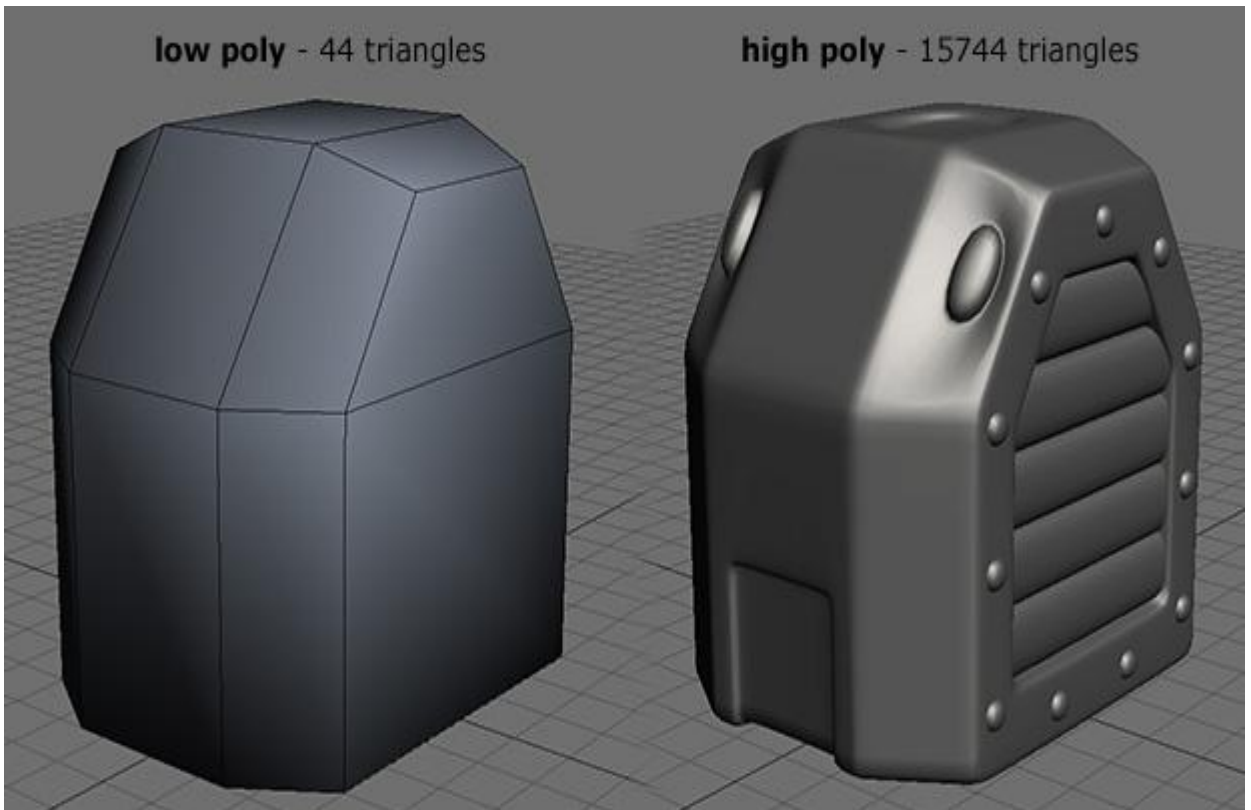


Рис. 1.4. Низькополігональна та високополігональна 3D моделі

Незважаючи на те, що полігональне моделювання вважається найпоширенішим способом побудови 3D об'єктів, це не єдиний метод, який використовується в створенні тривимірних зображень.

3D-скульптинг

3D-скульптура (яку також називають цифровою скульптурою) - це процес, під час якого художник ніби ліпить 3D-об'єкт на комп'ютері з матеріалу, схожого на оцифровану глину. Програмне забезпечення з пензлями і інструментами, які вдавлюють, тягнуть, стискають і згладжують, дозволяючи легко створювати деталізовані скульптури, які імітують реальні текстури і об'єкти.

Робота з 3D скульптурою подібна до полігонального моделювання тим, що вона також використовується для зміни форми моделі, але режим скульптури використовує зовсім інший робочий процес: замість того, щоб взаємодіяти з окремими елементами (вершинами, ребрами та гранями), область моделі змінюється за допомогою пензля (рис. 1.5.).

Іншими словами, замість вибору групи вершин, режим скульптури маніпулює геометрією в області впливу кисті.

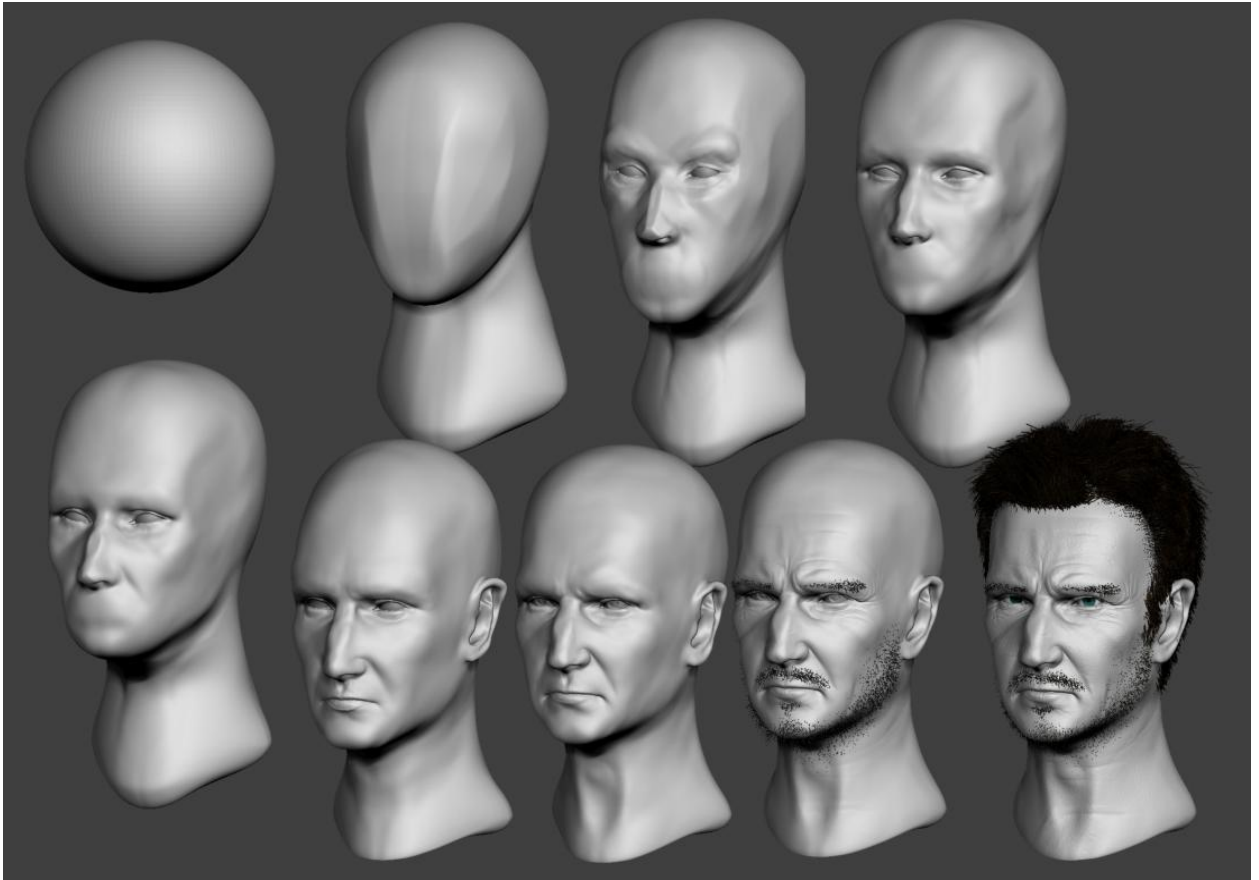


Рис. 1.5. Процес створення цифрової скульптури

Цифрова скульптура використовує багато комп'ютерних ресурсів та вимагає значної обчислювальної потужності.

З кожним підрозділом геометрії проект буде ставати все повільніше і повільніше, оскільки обробка вимагає компаундування. Найкраще цифрова скульптура показує себе під час створення органічних об'єктів, таких як персонажі або стовбури дерев.

Вивчення особливостей використання цифрової скульптури показало, що вона з успіхом застосовується в індустрії розваг, комп'ютерних іграх та рекламі.

NURBS

Ще одним з найбільш поширених методів моделювання являється NURBS - це аббревіатура Non-Uniform Rational B-Spline (нерівномірний раціональний Бі-сплайн), що описує математичну основу побудови кривих або сплайнів, використовуваних при моделюванні цього типу поверхонь.

Основою NURBS поверхні є крива. Поверхня створюється перетинанням кривих. Криві допомагають створювати і модифікувати поверхні. Таким чином створення і редагування кривих є важливою частиною NURBS моделювання.

Виробники програм для роботи з графікою використовують в своїх продуктах різні методи моделювання, що базуються на різних типах кривих. NURBS криві надають кращі властивості з усіх інших типів (рис. 1.6.).

З NURBS кривими можна помістити точку на кривій або поверхні в будь-якому місці, де це потрібно і змінити форму кривої або поверхні змінивши положення всього декількох контрольних точок, розташованих на або біля кривої.

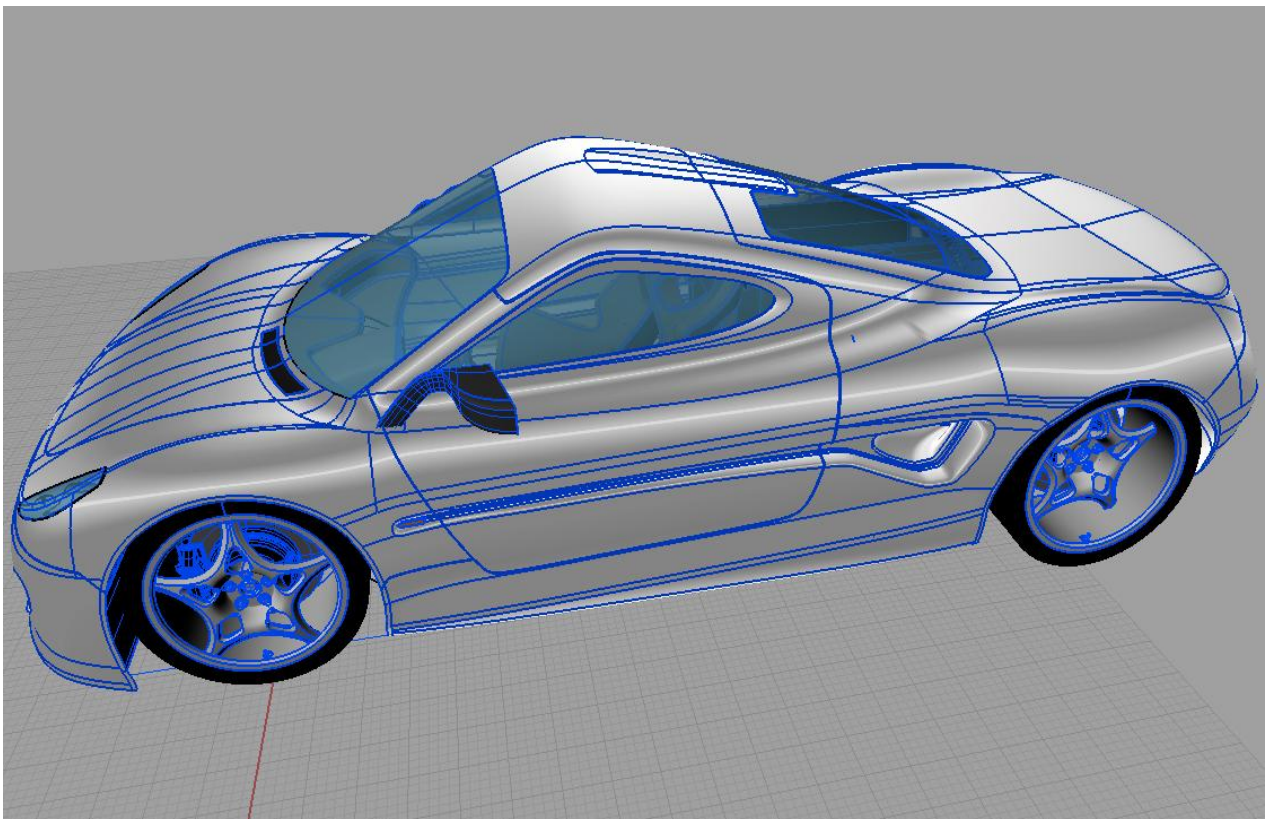


Рис. 1.6. Створена за допомогою NURBS модель

Параметричне моделювання - це процес моделювання з можливістю змінювати форму геометрії моделі, як тільки значення розміру змінюється.

Параметричне моделювання використовує комп'ютер для проектування об'єктів або систем, що моделюють атрибути компонентів із поведінкою у реальному світі (рис. 1.7.).

Параметричні моделі використовують інструменти дизайну на основі функцій, твердого та поверхневого моделювання для управління атрибутами системи.

Однією з найважливіших особливостей параметричного моделювання є те, що атрибути, пов'язані між собою, автоматично змінюють свої особливості.

Іншими словами, параметричне моделювання дозволяє дизайнеру визначати цілі класи фігур, а не лише конкретні екземпляри.

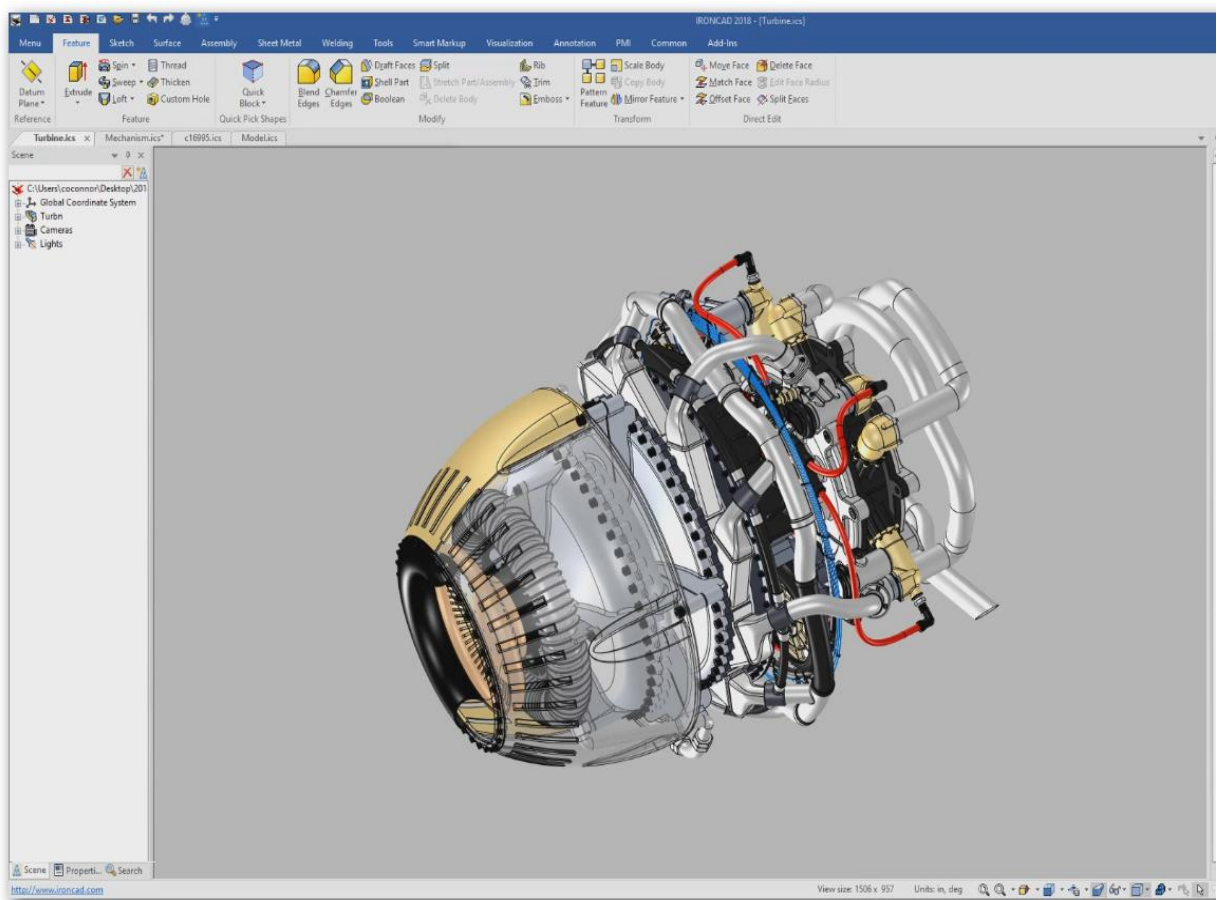


Рис. 1.7. Параметричне моделювання турбіни

Наприклад, щоб змінити форму виробу, дизайнеру довелося б змінити довжину, ширину і висоту. Однак при параметричному моделюванні потрібно змінити лише один параметр, інші два параметри налаштовуються автоматично. Отже, параметричне моделювання зосереджується на етапах створення форми і параметризує їх. Параметричні моделі будуються з набору математичних рівнянь.

Ідеї параметричного моделювання з'явилися ще на ранніх етапах розвитку САПР (система автоматизованого проектування), але якийсь час не могли бути здійснені через недостатню потужність комп'ютерних систем. Англomовний варіант назви - CAD, тобто Computer Aided Design.

Система автоматизованого проектування САПР - це програма, яка базується на двох основних підсистемах: проектування та обслуговування. За допомогою першої здійснюється сама побудова схем, креслень (рис. 1.8.). Друга служить для управління першою.

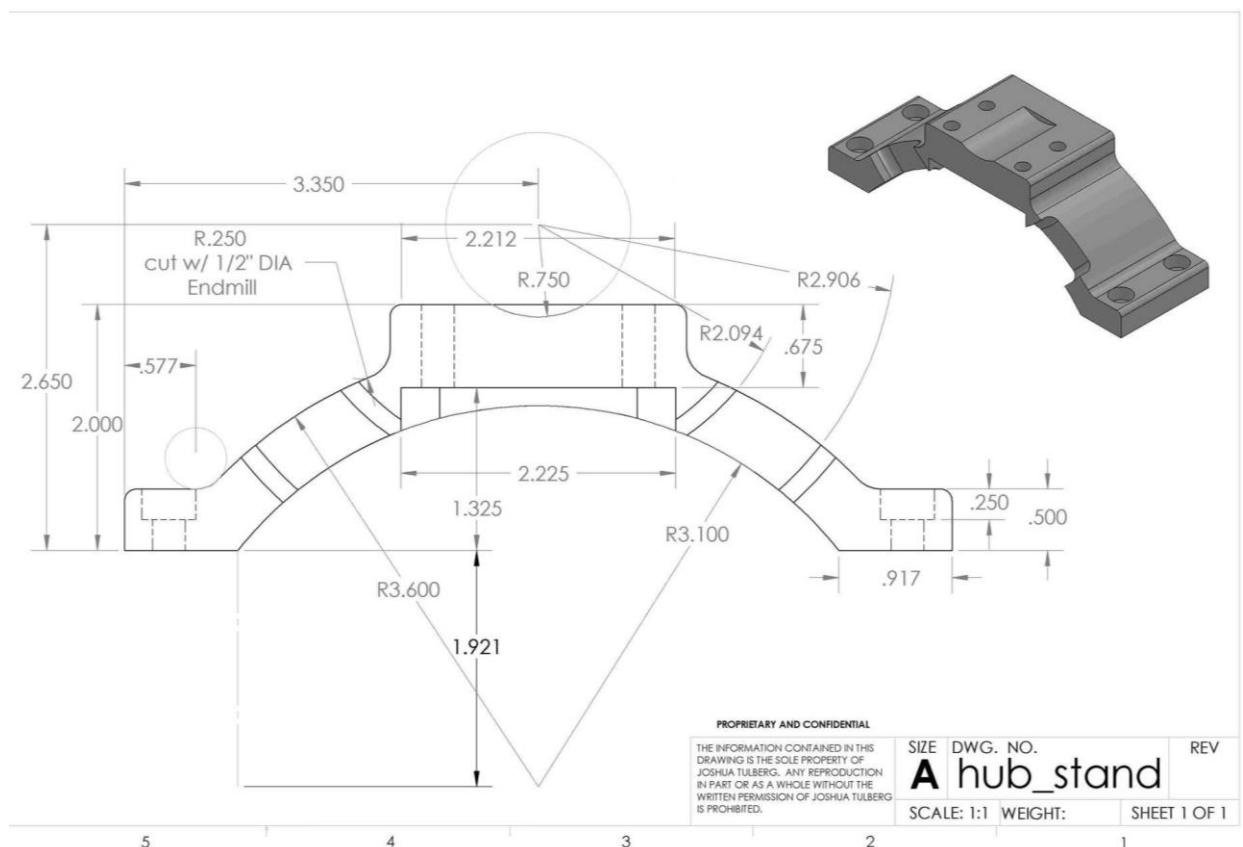


Рис. 1.8. Креслення та модель, створені за допомогою САПР

3D-сканування - це метод, який використовується для захоплення форми об'єкта за допомогою 3D-сканера.

В результаті виходить 3D-файл об'єкта, який можна зберегти, відредагувати і навіть надрукувати в 3D (рис. 1.9.). Існує безліч різних технологій 3D-сканування для сканування об'єктів, середовищ і людей. Кожна технологія 3D-сканування має свої обмеження, переваги та недоліки. 3D-сканери є потужними інструментами для професіоналів в декількох галузях, таких як автомобілебудування, авіація, стоматологія, ювелірні вироби, а також відеоігри, спецефекти і анімаційні фільми.

Одним з найпоширеніших методів 3D-сканування є фотограмметрія, також звана як 3D-сканування по фотографіях, відтворює 3D об'єкт з 2D-знімків за допомогою алгоритмів комп'ютерного зору і обчислювальної геометрії.



Рис. 1.9. Робота 3D-сканера

1.3. Програмне забезпечення для створення тривимірної графіки

Програмні засоби, що дозволяють створювати тривимірну графіку, тобто моделювати об'єкти віртуальної реальності і створювати на основі цих моделей зображення, дуже різноманітні (рис. 1.10.).



Рис. 1.10. Програмні засоби для роботи з 3D графікою

Останні роки стійкими лідерами в цій галузі є комерційні продукти, такі, як [14]:

Autodesk 3ds Max - професійне програмне забезпечення для 3D-моделювання, анімації і візуалізації при створенні ігор і проектуванні. В даний час розробляється і видається компанією Autodesk 3ds Max - це комп'ютерна програма, призначена тільки для комп'ютерної 3D-графіки, яка використовується для виробництва телевізійних і художніх фільмів, а також для архітектурної візуалізації та візуалізації продукції.

Додаток використовує як прямі маніпуляції, так і методи процедурного моделювання, а величезна бібліотека різних модифікаторів полегшує процес моделювання (рис. 1.11.).

3ds Max пропонує професійний набір інструментів. Тим не менш, студенти можуть отримати програмне забезпечення безкоштовно, і пробна версія також доступна впротягом 30 днів [18].

- Розробник: Autodesk
- Платформа: x86-64
- Остання версія: 2020 (Березень 2019 року)
- Створювані формати файлів: MAX (3ds Max)
- Мови програмування: C ++, C Sharp
- Модель ціноутворення: Підписка

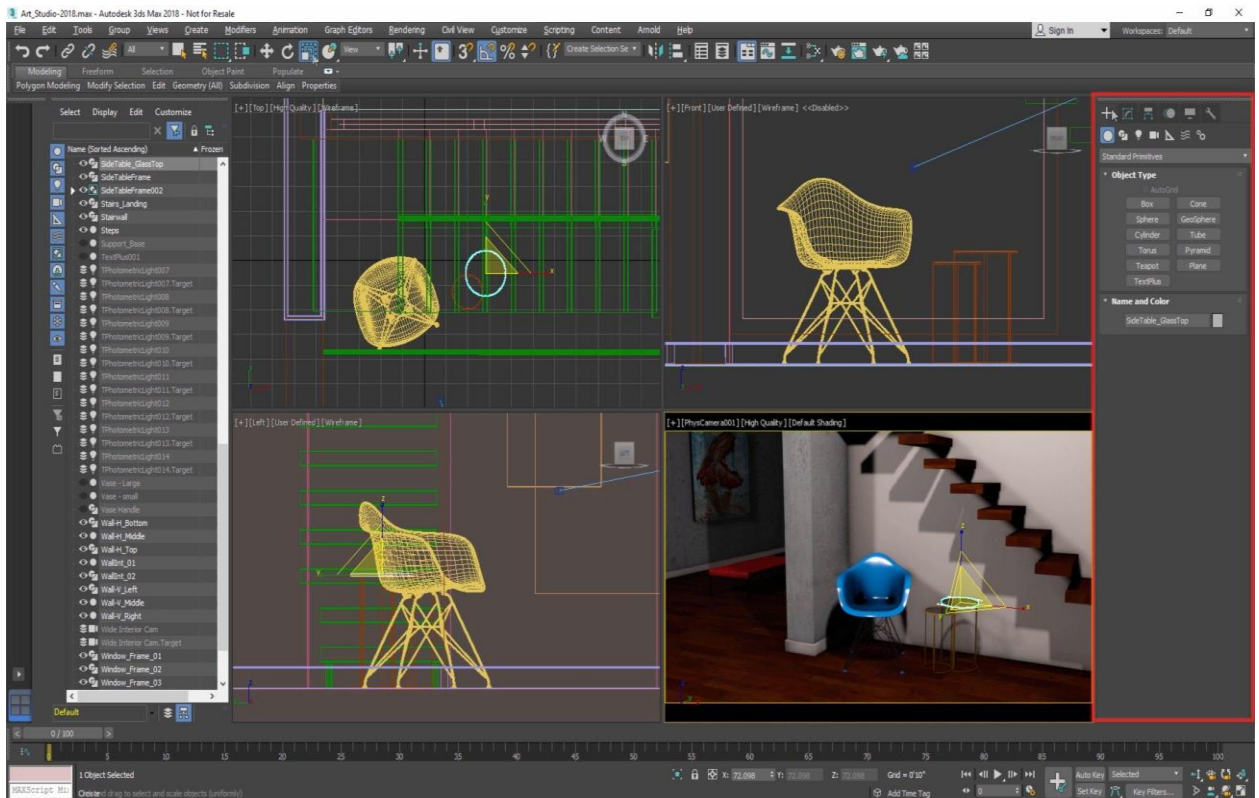


Рис. 1.11. Інтерфейс програми Autodesk 3ds Max

Autodesk Maya - редактор тривимірної графіки, доступний на Windows, macOS і Linux. Maya має промислову потужність з відповідною ціною. Maya

відмінно себе показує в моделюванні, текстуруванні, освітленні і рендерінгу - її великий набір функцій включає в себе частинки, волосся, фізику твердого тіла, тканини, симуляції рідини і анімацію персонажів (рис. 1.12.).

Цей рівень потужності також відповідно коштує - передплата на Майя не дешева. Майя має одні з найкращих інструментів 3D, що наявні на ринку, і це надійне інвестування.

Маюа названа в честь санскритського слова **मया** māyā, майя, що означає «ілюзія».

- Розробник: Autodesk, Inc
- Автор-творець: Alias Systems Corporation
- Остання версія: 2019.2 (31 липня 2019)
- Написано на: C ++, Maya Embedded Language, Python, C Sharp
- Модель ціноутворення: Підписка

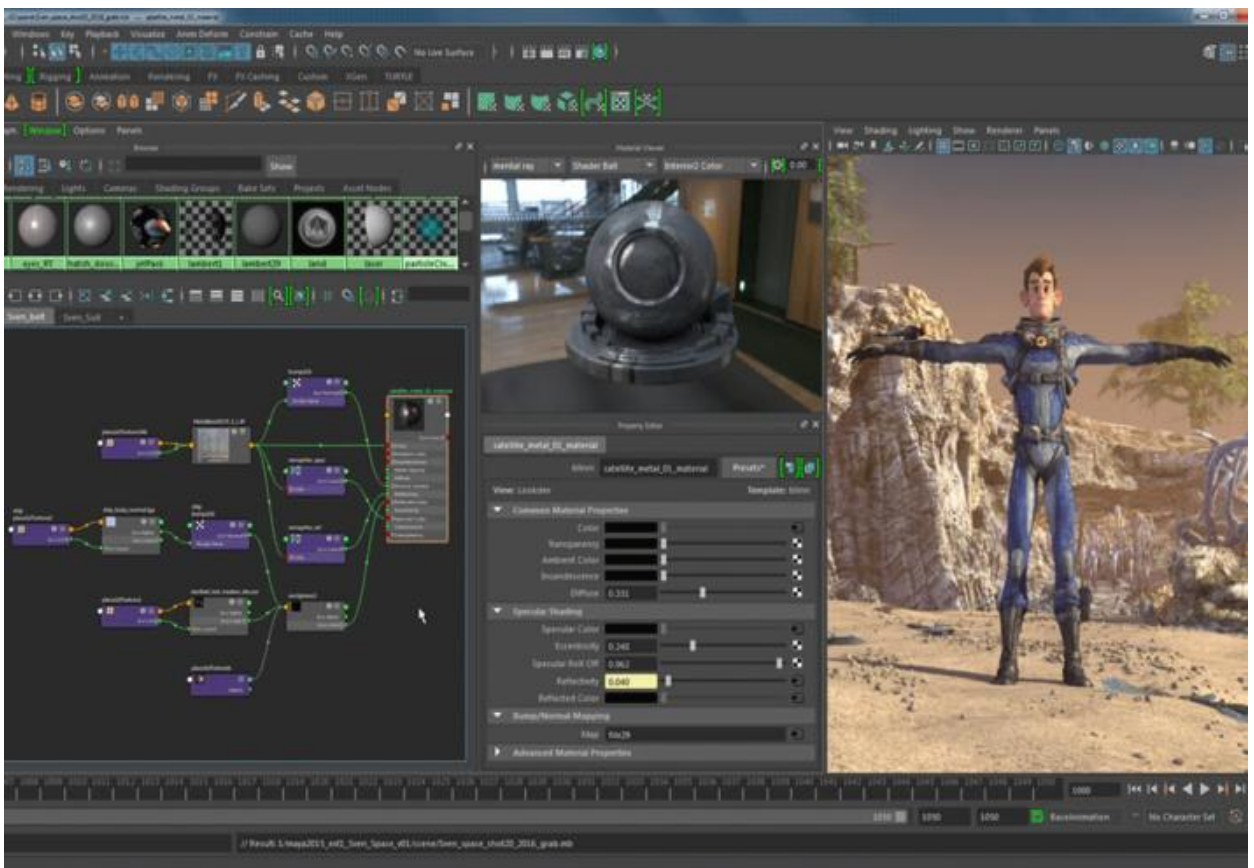


Рис. 1.12. Інтерфейс програми Autodesk Maya

Blender 2.8 це безкоштовне програмне забезпечення з відкритим кодом для роботи з комп'ютерною графікою з функціоналом професійного рівня. Blender - це додаток для моделювання, текстурування, анімації та візуалізації. Довгоочікувана версія 2.8 (запущена в липні 2019 року) надає користувачу сучасний, інтуїтивний інтерфейс, інтерактивне відображення в режимі реального часу, безліч виправлень та нових функцій [6].

Попередні версії Blender часто критикували за те, що робота з ним для нових користувачів не завжди очевидна.

Особливо незадоволені були ті, хто переходив з інших пакетів для роботи з 3D-графікою (Max, Maya і інших), оскільки принципи роботи і основні гарячі клавіші в Blender багато в чому не відповідали прийнятим в індустрії стандартам.

Так, при розробці Blender 2.8 була пророблена робота по поліпшенню користувацького досвіду. У тому числі багато речей були приведені до загальноприйнятих стандартів (рис. 1.13.).

Серед основних цілей цих нововведень було і зменшення порогу входу для нових користувачів.



Рис. 1.13. Інтерфейс програми Blender 2.8

Шаблони та робочі простори дозволяють швидко розпочати роботу з такими завданнями, як ліплення, малювання текстур або відстеження руху. Їх можна налаштувати для створення власного ефективного робочого середовища.

Eevee - це новий рендерер в реальному часі, заснований на фізиці. Він працює як в якості засобу рендеринга для остаточних кадрів, так і в якості движка, керуючого реальним вікном перегляду Blender для створення ресурсів.

Він має розширені функції, такі як об'ємні показники, відображення і заломлення в просторі екрану, підповерхневе розсіювання, м'які і контактні тіні, глибина різкості, розмиття руху камери і колірна корекція.

Матеріали в Eevee створюються з використанням тих же шейдерних нодів, що і в Cycles, що спрощує рендеринг існуючих сцен [5]. Для користувачів Cycles це робить Eevee чудовим рендером, який працює в режимі реального часу, та як передперегляд (рис. 1.14.).



Рис. 1.14. Сцена, візуалізована за допомогою Eevee

- Розробник: Blender Foundation
- Дата виходу: 1998 г.
- Автор-творець: Тон Розендаль
- Остання версія: 2.81 (21 листопада 2019);
- Написано на: C ++, Python

ZBrush 2019 це автономний додаток для ліплення та моделювання, який найкраще підходить для створення органічних форм, хоча останні оновлення поступово покращували його здібності роботи з твердими поверхнями [19].

ZBrush створений не тільки для ліплення: він також може бути використаний для створення UV розгортки та фарбування текстур, що дозволяє досвідченим художникам виготовляти цілі фігури, одяг та реквізит, готові до візуалізації.

ZBrush широко використовується в дизайні концептів, створенні кіно, комп'ютерних ігор, іграшок, колекційних моделей, наукових ілюстрацій і в багатьох інших сферах також має інструменти, спеціально спрямовані на 3D-друк.

Відмінною особливістю даного програмного засобу є імітація процесу «ліплення» тривимірної скульптури, посиленого движком тривимірного рендеринга в реальному часі, що істотно спрощує процедуру створення необхідного тривимірного об'єкту (рис. 1.15.).

- Розробник: Pixologic
- Остання версія: 2019
- Дата виходу: 1999
- Ліцензія: Пропріетарна
- Операційна система: Microsoft Windows, Mac OS
- Тип: Пакет для 3D, 2,5D малювання



Рис. 1.15. Інтерфейс програми ZBrush 2019

Cinema 4D R20 - програмне забезпечення для 3D моделювання як для початківців, так і для професіоналів. Махон's Cinema 4D існує вже багато років і його високо цінують у сфері графіки руху, візуалізації та ілюстрації. Це професійне, складне програмне забезпечення, відоме своєю загальною стабільністю та тим, що це програма CG з найпростішою кривою навчання [17].

Набір інструментів параметричного моделювання Cinema 4D має багатий функціонал а також має можливість додати ще більше функціональних можливостей за допомогою ряду плагінів. Останній випуск також представив об'ємне моделювання.

Cinema 4D або скорочено C4D фірми Махон є пакетом для створення тривимірної графіки і анімації (рис. 1.16.). Cinema 4D є універсальною комплексною програмою для створення і редагування тривимірних ефектів і об'єктів. Дозволяє рендерити об'єкти по методу Гуро.



Рис. 1.16. Інтерфейс програми Сінема 4D

Cinema 4D надає велику кількість опцій для недеструктивної роботи: параметричне моделювання, процедурні шейдери, текстури і інше. А система дублів Cinema 4D дозволяє зберігати і керувати безліччю версій сцени в одному єдиному файлі.

Крім вбудованого рендеру Cinema 4D може працювати і зі сторонніми рендерами, як вбудованими безпосередньо в саму середу програми, так і за допомогою конекторів. Частина з сторонніх рендерів до R19 безпосередньо підтримувалися через вбудовані пакети Cinema 4D Visualize, Cinema 4D Studio і BodyPaint 3D конектор CineMan.

- Розробник: MAXON Computer GmbH
- Читаються формати файлів: CINEMA 4D
- Остання версія: 21
- Дата виходу: 1990
- Операційна система: 100 macOS, Microsoft Windows

Houdini від SideFX це професійний програмний пакет для роботи з тривимірною графікою, розроблений компанією Side Effects Software.

Головна відмінність даного пакета в тому, що він є середовищем візуального програмування. Широко використовується в галузі візуальних ефектів для створення 3D-зображень, процедурний підхід на основі вузлів Houdini забезпечує цифровим художникам високий рівень потужності, гнучкості та контролю (рис. 1.17.).

Цей вузловий робочий процес подобається не всім, але Houdini також має більш традиційні інструменти для прямої взаємодії з полігонами на екрані [15].

Як і у Maya, цей рівень потужності та нестандартний робочий процес можуть бути складними для початківців. SideFX пропонує Houdini Apprentice, безкоштовну версію Houdini FX, яку можуть використовувати студенти, художники та любителі для особистих некомерційних проєктів.

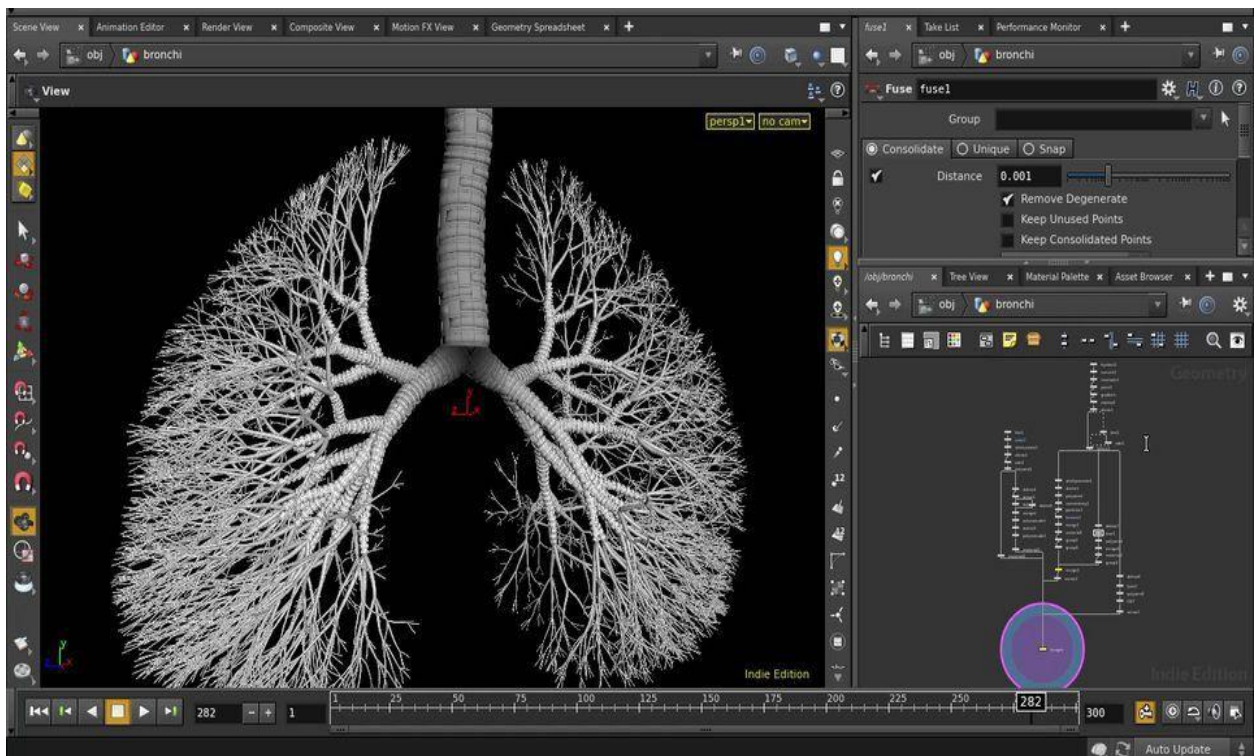


Рис. 1.17. Інтерфейс програми Houdini FX

Безкоштовна версія надає доступ до практично всіх функцій повноцінного Houdini FX для розвитку своїх навичок та роботи над особистими проектами. Повнофункціональний Houdini Indie також пропонує доступний комерційний варіант для невеликих студій.

Хоча Houdini в основному використовується для динамічних середовищ та ефектів частинок, він включає повний набір інструментів для художників, які хочуть використовувати його для інших областей, таких як моделювання, анімація або рендеринг.

- Розробник: Side Effects Software
- Написано на: C ++
- Остання версія: 17.5 (2 019)
- Операційна система: Microsoft Windows, OS X, Linux
- Перший випуск: грудень 1996

Висновки до розділу 1

Завдяки безперервному та швидкому розвитку технологій, сьогодні 3D-рендерінг - це процес, який дозволяє створювати фотореалістичні зображення за допомогою використання спеціалізованого програмного забезпечення.

3D-моделі використовуються у різних сферах, включаючи архітектуру, інженерію, ілюстрації, комерційну рекламу, відеоігри, фільми та анімаційні мультфільми. У більшості випадків використання 3D-графіки дозволяє знизити витрати в порівнянні зі зйомкою фото або відео.

3D-моделювання - невід'ємна частина багатьох творчих спеціальностей. Інженери та архітектори використовують його для планування та проектування своєї роботи. Аніматори та дизайнери ігор використовують 3D-моделювання для втілення в життя своїх ідей.

Оцінивши популярні програмні засоби для роботи з тривимірною графікою можна зробити такі висновки: усі вони загалом мають великий функціонал та вже реалізовані базові функції, які допомагають створювати комп'ютерні візуалізації, цифрові скульптури, 3D моделі та їх анімації.

РОЗДІЛ 2

ФІЗИЧНЕ МОДЕЛЮВАННЯ В КОМП'ЮТЕРНІЙ ГРАФІЦІ ТА ЙОГО РЕАЛІЗАЦІЯ В ПАКЕТІ HOUDINI

2.1. Пакет процедурного моделювання Houdini

Houdini - це пакет процедурного моделювання, анімації, ефектів, симуляції, рендеру і композитингу. Для того, щоб зрозуміти, як працювати з Houdini, треба зрозуміти основні принципи процедурного моделювання.

Процедурне моделювання - це загальний термін для позначення ряду методів в комп'ютерній графіці для створення тривимірних моделей і текстур з наборів правил [15]. L-системи, фрактали і генеративне моделювання є методами процедурного моделювання, оскільки вони застосовують алгоритми для створення сцен. Набір правил може бути вбудований в алгоритм, налаштовуватися параметрами або набором правил.

Хоча всі методи моделювання на комп'ютері вимагають алгоритмів для управління і зберігання даних в якийсь момент, процедурне моделювання фокусується на створенні моделі з набору правил, а не на редагуванні моделі за допомогою призначених для користувача спеціальних команд.

Процедурне моделювання часто застосовується в тих випадках, коли створення тривимірної моделі з використанням класичних методів 3D-моделювання було б занадто громіздким або коли потрібні більш спеціалізовані інструменти. Це часто стосується створення рослин, архітектури або ландшафтів.

Усі ці принципи лежать в основі роботи з Houdini. Прабатько Houdini - це програмне забезпечення Prisms, по суті складається з декількох блоків, що відповідають за виконання різних задач.

Офіційна історія Houdini починається саме з дати виходу Prisms з графічним інтерфейсом в 1987 році [2].

Уже тоді ПО працювало по принципам побудови нодових мереж (рис. 2.1.).

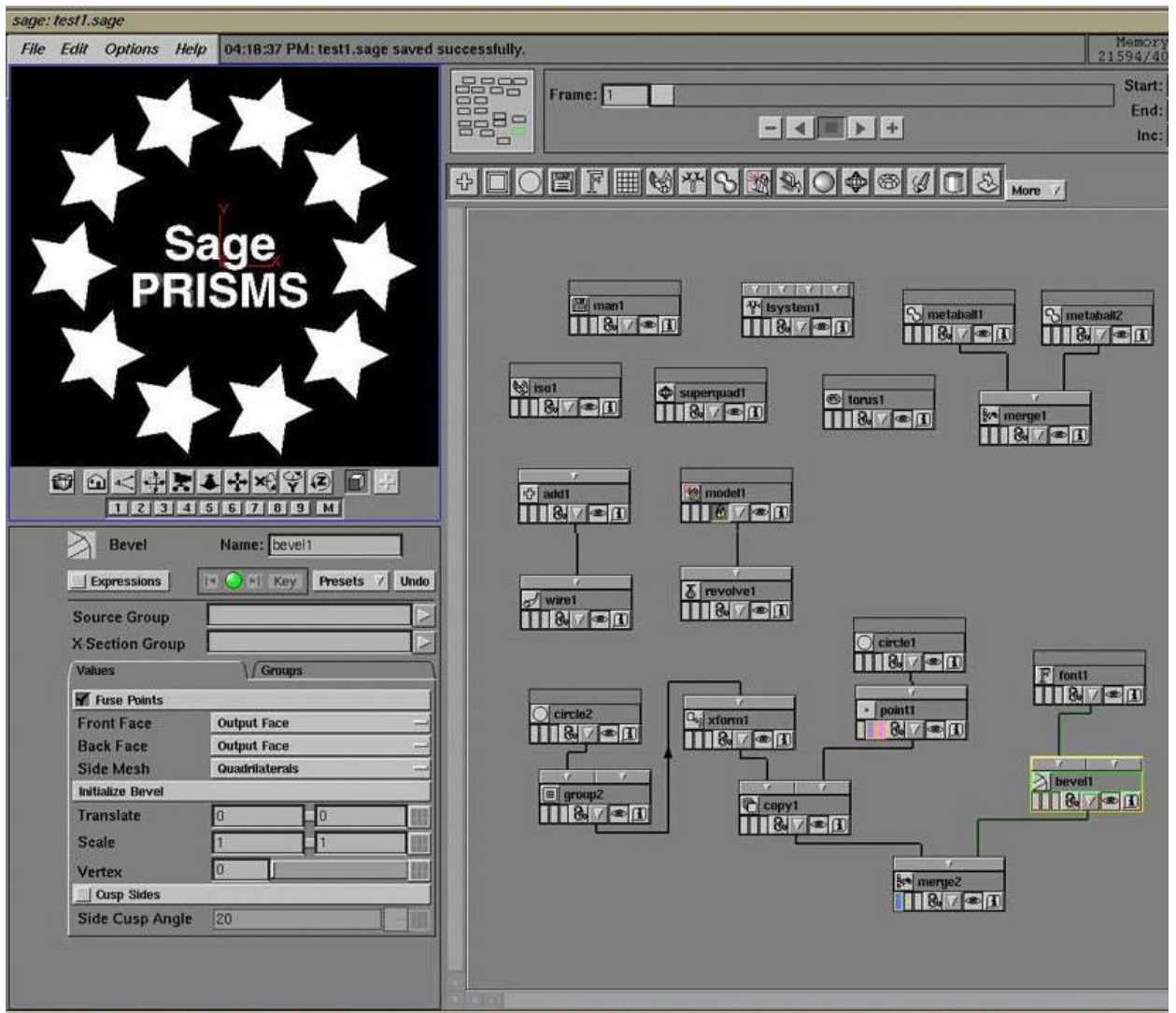


Рис. 2.1. Інтерфейс програми Prism

Як і раніше вся робота в Houdini будується саме на нодовій структурі (зв'язках вузлових точок), які розташовуються в особливому просторі, утворюючи ланцюг послідовностей. До кожного кроку послідовності можна повернутися і внести зміни, які одразу вплинуть на всі наступні кроки послідовності нодів.

Ідея створення Houdini полягає в тому, щоб створити графічний пакет з візуальними програмними вузлами. Тобто візуальне програмування геометрії. Це відрізняє Houdini від більшості 3D-пакетів, де початкова логіка була іншою.

Розглянемо процес побудови геометрії на простому прикладі, без графічної оболонки, яка є в Houdini, Maya або будь-якому іншому пакеті.

Опишемо куб 6 гранями. Для цього визначимо у світовому просторі координату кожної з 8 його вершин, з'єднуючи їх ребрами і заповнюючи полігонами. На кожній грані зробимо отвір. Якщо у певний момент потрібно зробити, щоб це був не куб, а октаедр з 8 гранями, тоді потрібно повернутися до перших рядків коду, переписати їх і наступні рядки, тому що змінюється координата кожної грані [11].

Houdini зберігає вищеописану послідовність дій в ланцюзі з'єднаних між собою нод, які створюються автоматично. Уявіть собі блокнот, який сам переписує всі наступні рядки, якщо змінюється попередня. Тільки рядки ці відображаються в графічному вигляді нод і зв'язку між ними можна визначати по-своєму. Тепер можна просто повернутися до кроку визначення граней, внести зміни і всі вони передадуться на наступні кроки, тобто отвір з'явиться на всіх гранях.

Такий підхід до побудови форм називається процедурним. Тобто Houdini - це процедурний пакет роботи з графікою. Під цей підхід реалізований весь інтерфейс, і логіка будується саме з опорою на процедурність. Таким чином, інтерфейс Houdini ділиться на елементи, які беруть участь в операціях з логікою зв'язків між нодами.

Нодові мережі Houdini схожі на файлову систему комп'ютера, де мережі схожі на папки, а вузли - на файли.

Вузли можуть бути підмережами, які містять інші вузли. Коренева мережа містить кілька готових мереж, таких як / geo (мережа «сцени») і / out (куди йдуть вузли рендеринга)

Мережі мають «тип» (наприклад, «геометрична мережа» або «динамічна мережа»), який визначає, які типи вузлів ви можете помістити в мережу (рис. 2.2.).

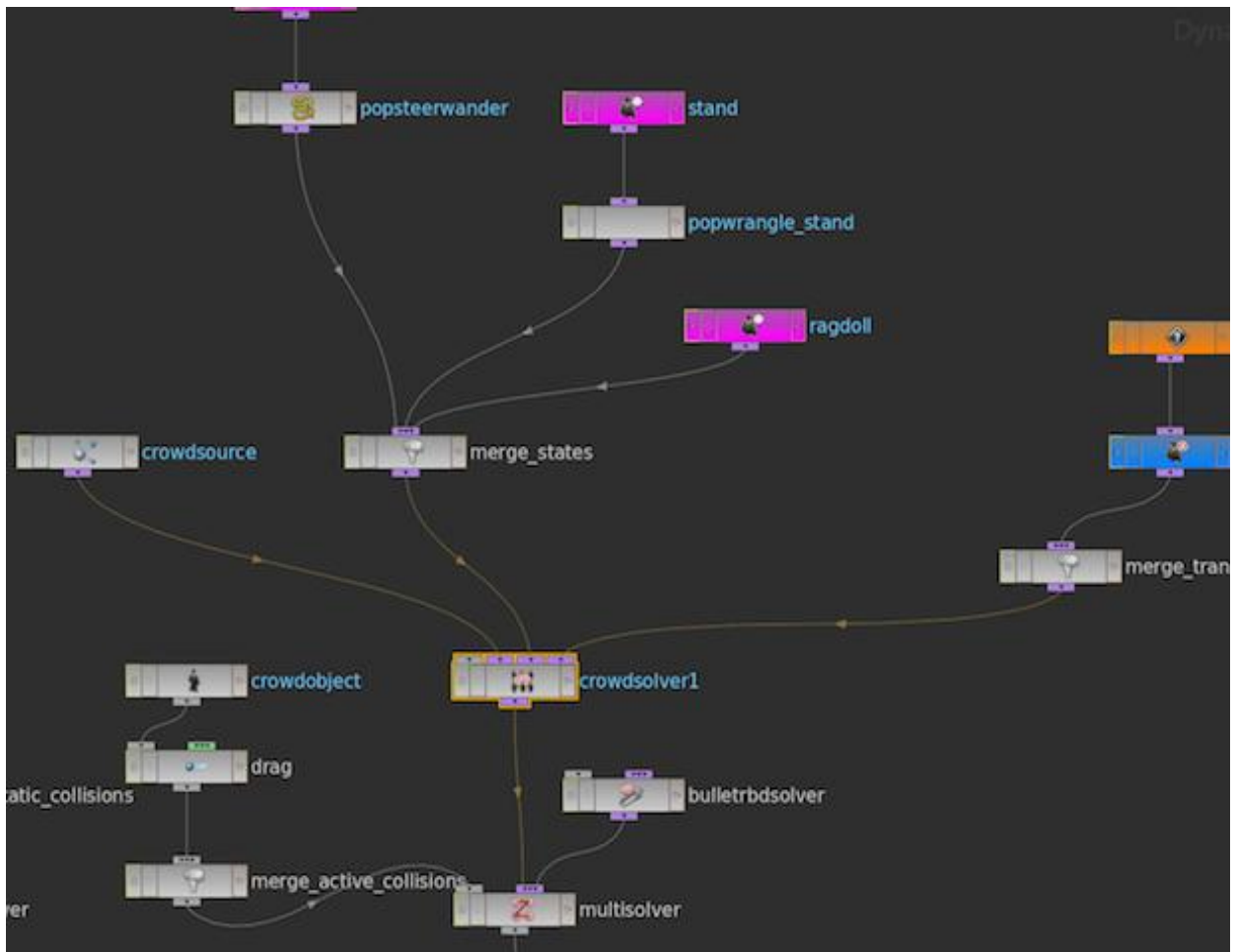


Рис. 2.2. Система нодів Houdini

Рівень сцени (/ geo) містить такі об'єкти, як персонажі, реквізит і джерела світла. Об'єкти геометрії - це підмережі, які містять вузли геометрії, які визначають геометрію об'єкта. Цей дворівневий дизайн відрізняється від інших пакетів, таких як Maya і 3D Studio Max, де всі вузли знаходяться на одному рівні.

Рівень сцени також може містити вузли, які не уявляють об'єкти на сцені, вони є чисто підмережами. Наприклад, коли ви налаштуєте симуляцію, вузли, що визначають симуляцію, знаходяться всередині підмережі Dynamics на рівні сцени.

Інтерфейс Houdini розділений на різні типи панелей, такі як 3D-переглядач, мережевий редактор і редактор параметрів.

Кожна панель може бути орієнтована на окрему мережу, але за замовчуванням всі вони налаштовані так, щоб слідувати тій же мережі, що й інші.

Кожен вузол має користувальницький інтерфейс налаштувань, званий параметрами, які можна редагувати в панелі редактора параметрів, щоб змінити функцію вузла. Між вікнами є пунктирні смужки. Натискаючи на них можна міняти положення вікон місцями (рис. 2.3.).

Стрілки між вікнами згортають або розгортають панелі на весь екран. Чорні стрілки в кутку панелей - дозволяють тимчасово згорнути або показати вікно.

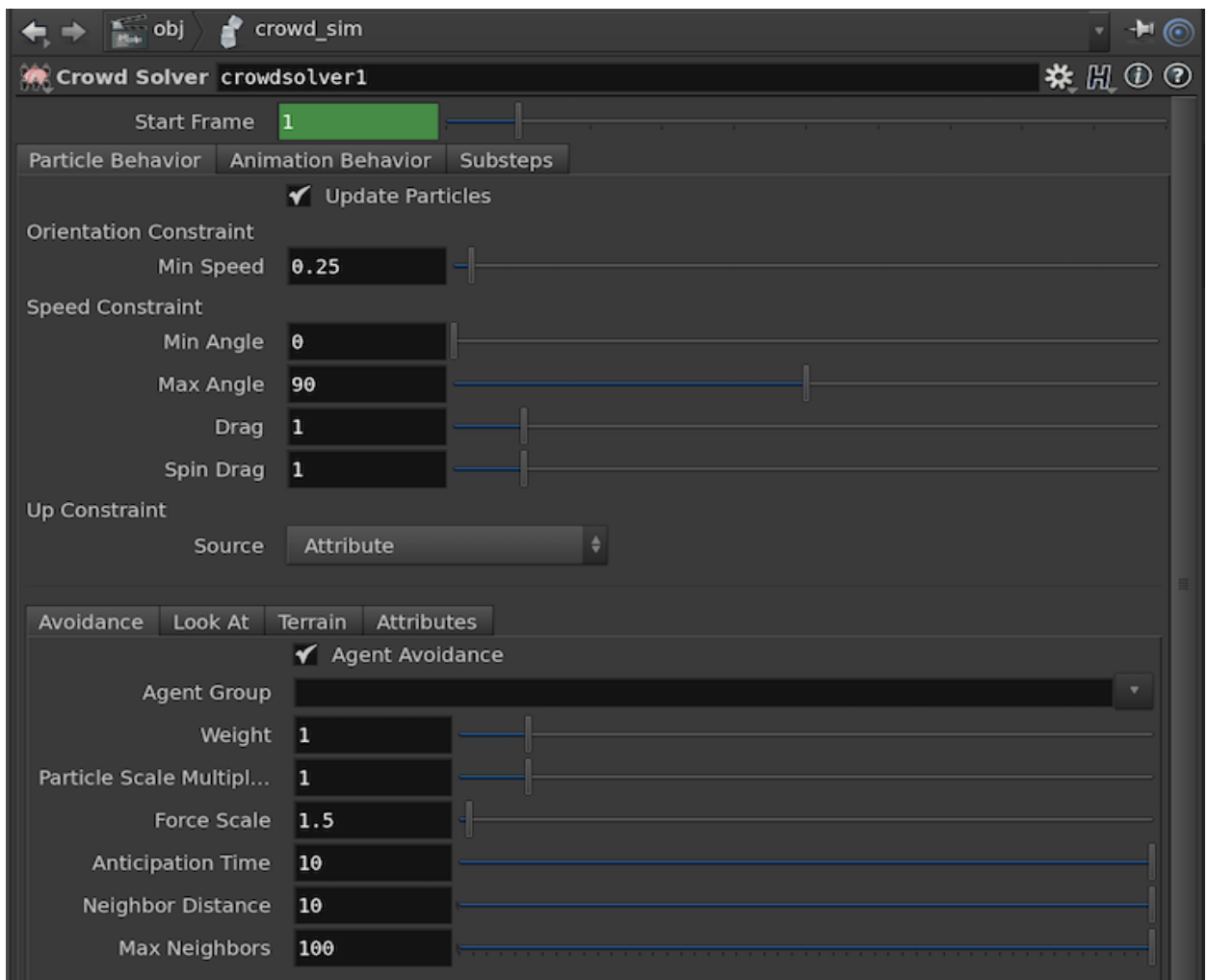


Рис. 2.3. Вікно налаштування параметрів

Після наведення в просторі між вікнами (помаранчеве підсвічування) - курсор набуває вигляду обопільної стрілки і дозволяє змінювати розмір вікна перетягуванням.

Вузли або ноди є будівельними блоками сцени. Сцена Houdini побудована з вузлів, організованих в мережі. Різні типи мереж управляють різними частинами Houdini.

Використання інструментів полки автоматично створює вузли. Наприклад, коли ви натискаєте інструмент Vox на вкладці «Створити полку», Houdini створює новий контейнерний об'єкт Geometry з вузлом Vox всередині.

Houdini підтримує різні типи геометрії, такі як багатокутники, NURBS і Безьє, геометричні примітиви, такі як кола і сфери, метаболли. Houdini називає кожен «шматок» геометрії будь-якого з цих типів примітивом. Наприклад, грань полігону є примітивом, полігональна крива - примітивом, метабол - примітивом. Це не ідеальна термінологія, тому що деякі "примітиви" мають редаговані компоненти. Наприклад, межі полігону мають точки і вершини.

Велика частина інформації, що описує сцену, зберігається в атрибутах, приховані дані зберігаються в моделях, примітивах, точках і вершинах.

Деякі вузли можуть містити інші вузли. Наприклад, об'єктний об'єкт Geometry Container існує на рівні сцени. Він містить мережу поверхневих вузлів, які визначають геометрію об'єкта [11].

Кожен вузол в мережі виконує певну функцію. У мережах геометрії, композитінга і СНОР кожен вузол створювати чи змінювати дані, що проходять через вузол.

У Houdini є кілька різних методів моделювання рідин, таких як вода, багно, лава та інших, кожен з яких має свої сильні та слабкі сторони. Моделювання системи часток для відображення рідини використовує частки для представлення різних типів рідин.

Для візуалізації і рендерінга частки спливають, тобто поверхня створюється на основі системи частинок (рис. 2.3.).

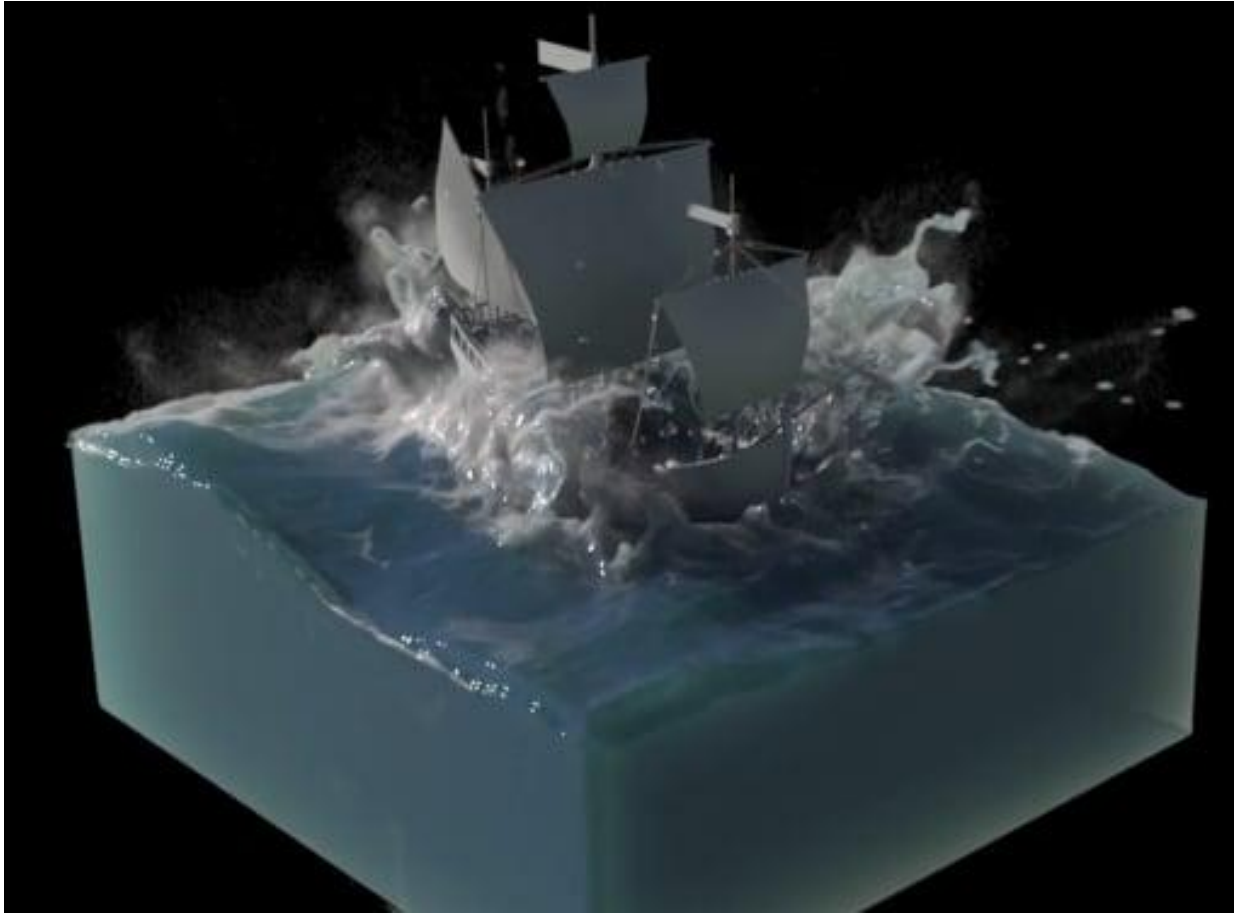


Рис. 2.4. Приклад симуляції рідини в Houdini

Для того, щоб отримати рідину більш високої роздільної здатності, треба зменшити параметр розділення частинок на текучому об'єкті. Значення розділення за замовчуванням досить високе, щоб видавати швидкі результати при прототипуванні, але потрібно знизити його для отримання більш високої якості, доки не буде досягнуто потрібного ефекту [12].

Роздільна здатність частинок вимірюється в одиницях сцени Houdini, які відповідають метрам у динамічному світі Houdini. Отже значення поділу частинок 0,1 становить 10 см.

Для симуляцій вогню та інших схожих явищ використовується спеціальні вузли мережі Pyro. Вони посилаються на об'ємний пакет симуляції рідини Houdini.

Руго використовується для фіксації руху вогню, диму та вибухів, але також може бути використаний для імітації інших явищ, подібних до рідини, таких як туман або лавина (рис. 2.4.).

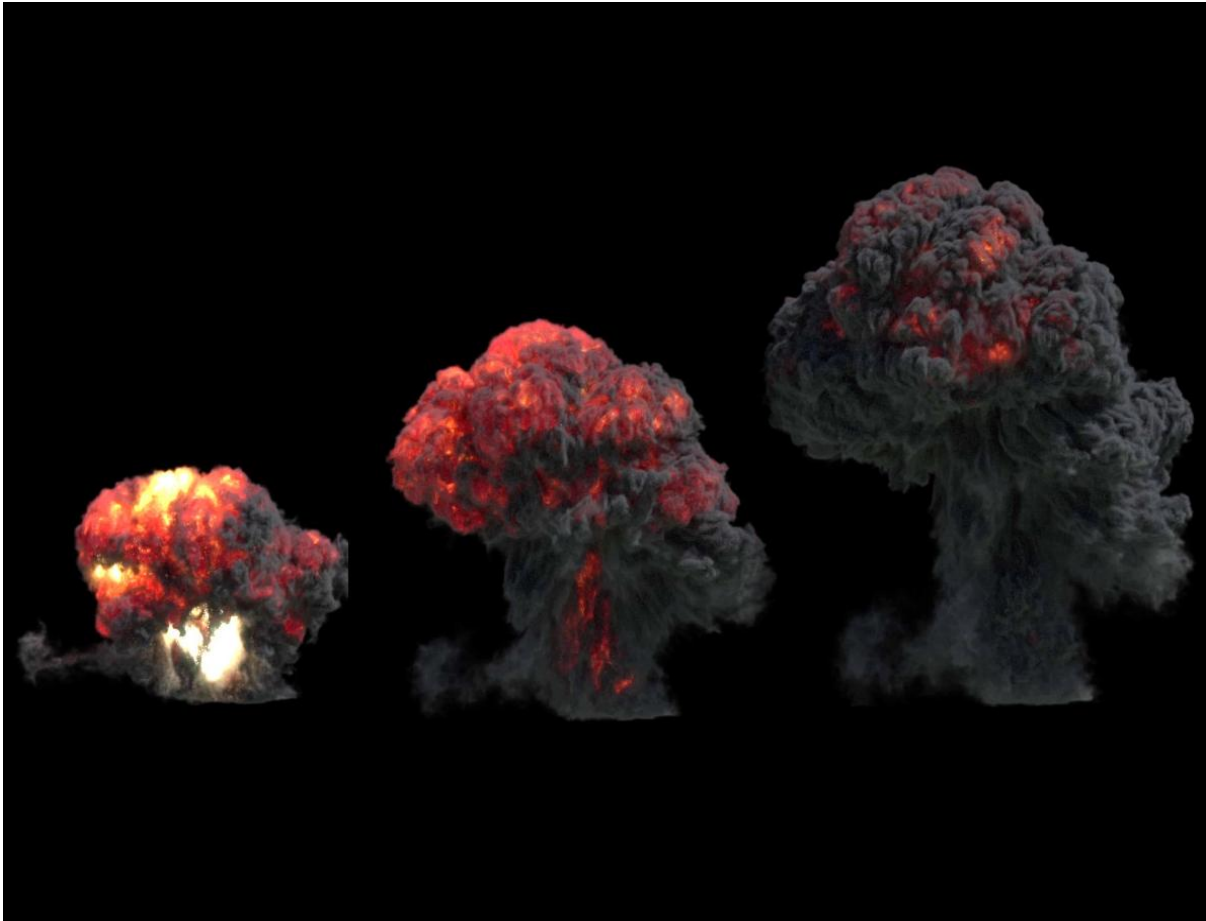


Рис. 2.4. Симуляція вибуху в Houdini Pyro

Основними складовими мережі динамічної пірографії є об'єкт (Object), джерело (source) та вирішення (Solver) [15]. Об'єкт моделювання з усіма необхідними полями створюється вузлом Smoke Object (Sparse).

Вузол джерела відповідає за імпорт джерел та об'єднання їх із відповідними імітаційними полями. Нарешті, Pyro Solver піклується про застосування сил і еволюцію полів рідини. Sourcing відповідає за збільшення об'єму симуляції.

Наприклад, можна постійно додавати сажу і вогонь всередину димоходу, щоб відтворювати дим, що піднімається з нього.

Інструменти на полиці Hair Utils дозволяють швидко налаштувати волосся і хутрянні об'єкти, а також анімувати і моделювати. Інструменти на вкладках Guide Process і Guide Brush використовуються для укладання хутра [10].

Основна відмінність полягає в тому, що інструменти на полиці Guide Process дозволяють впливати на розміщення і орієнтацію напрямку волосся, маючи атрибути шкіри, тоді як інструменти на полиці Guide Brush використовують фактичні інструменти для розгладжування, щоб створювати волосся і стилізувати його (рис. 2.5.).



Рис. 2.5. Використання системи хутра в Houdini

Щоб створити хутро, треба почати з додавання геометрії і використання інструменту «Додати хутро» на полиці для волосся. За замовчуванням створюються два вузли: Guide Groom і Hair Generate (рис. 2.5.).

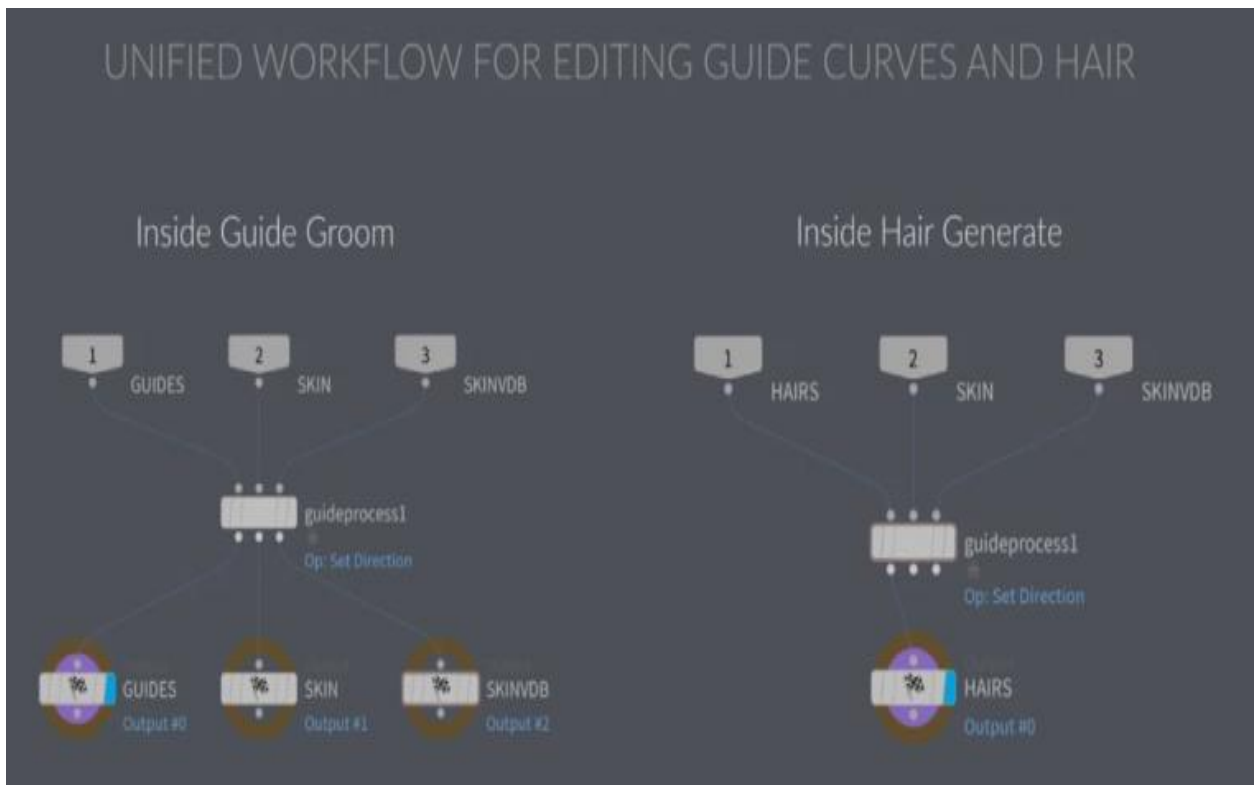


Рис. 2.6. Вузли Guide Groom і Hair Generate

Вузли Guide Groom і Hair Generate мають однакові налаштування та однаковий робочий процес, за винятком того, що мережа для генерації волосся має менше результатів. Це робить його дуже простим у використанні, оскільки все робиться в СОП з кривими.

Вузол Guide Guide дозволяє промаркувати багато різних операцій і дуже легко контролювати, де операція відбувається по кривій, за допомогою управління атрибутами на шкірі. Це дозволяє дуже швидко створити волосся та хутро природного вигляду (рис. 2.7.).

Всі операції протікають через один графік. Всі компоненти, що входять до складу мережі, відображаються в єдиній системі.

Кожен з вузлів також має щонайменше 3 входи та 3 виходи, що дозволяє пропускати через вузли 3 типи даних. Це призводить до дуже чистої та зручної для читання мережі та позбавляє від необхідності посилатися на інші вузли.



Рис. 2.7. Симуляція різних типів хутра в Houdini

Ще одна перевага полягає в тому, що операторів можна розмістити в будь-якому порядку, що надає повний контроль та гнучкість.

Генерування хутра без будь-яких напрямних корисно для створення таких речей, як трава або дуже коротке хутро. Це економить значну кількість часу на обчислення.

Часто при моделюванні твердого тіла потрібно, щоб твердий об'єкт розколовся на частини у наслідок деякого удару або використання сили. Наприклад, потрібно, щоб землетрус зруйнував будинок, через руйнування

бетонних стін, руйнування дерев'яних дверей і руйнування скляних вікон (рис. 2.8.).



Рис. 2.8. Симуляція руйнування об'єктів в Houdini

Більшість інструментів для розриву геометрії в Houdini підтримують робочий процес попереднього розбиття, коли геометрія розбивається на шматки, при цьому шматочки з'єднуються обмеженнями склеюванням [11].

Попереднє розбиття дає повний художній контроль над виглядом руйнування (наприклад, потрібні великі блокадні шматки чи маленькі зазубрені шматочки). Об'єкт розсіплеться, коли прикладена сила перевищить міцність склеювання.

2.2. Фізичне моделювання в комп'ютерній графіці

Створення анімаційних інструментів для плавних рухів є важливим і складним завданням для багатьох додатків комп'ютерної графіки. Використання заснованих на фізиці моделей для потоку рідини може дуже допомогти в створенні таких інструментів.

Фізичні моделі, на відміну від ключових кадрів або процедурних методик, дозволяють аніматору майже без особливих зусиль створювати цікаві, що кружляють, схожі на рідину поведінки [12]. Крім того, взаємодія потоків з об'єктами і віртуальними силами обробляється елегантно. До недавнього часу вважалося, що моделі фізичної рідини занадто дорогі, щоб забезпечити взаємодію в реальному часі.

Багато в чому це було пов'язано з тим, що в попередніх моделях використовувалися нестабільні схеми для вирішення фізичних рівнянь, які керують рідиною.

У контексті фізики слово "рідина" може означати щось інше, ніж ви зазвичай можете думати. У фізиці рідини розпадаються на дві частини категорії нестисливого і стисливого потоку. Нестисливий потік - це рідина, наприклад вода або сік.

Стисливий потік, з іншого боку, відповідає таким газам, як повітря або пара. Стисливий потік називається стисливим, тому що можна легко змінити об'єм цієї рідини. Треба зауважити, що не існує такого поняття, як 100-відсотково нестислива рідина. Всі рідини, навіть вода можуть певною мірою змінювати об'єм.

Існує багато, багато способів моделювання рідин. У графіці найпоширеніші дві методики - це симуляції на основі сітки та моделювання на основі частинок.

Моделювання на основі сітки, як правило, є високоточними, хоча й відносно повільними порівняно з розчинами на основі частинок [20].

Моделювання на основі частинок, як правило, набагато швидше, але вони, як правило, виглядають не так добре, як симулятори на основі сітки.

Спочатку треба описати загальне рівняння нестисливих рідин Нав'є-Стокса:

$$\frac{\partial \vec{u}}{\partial t} + \vec{u} \cdot \nabla \vec{u} + \frac{1}{\rho} \nabla p = \vec{F} + \nu \nabla \cdot \nabla \vec{u} \quad (1)$$

$$\nabla \cdot \vec{u} = 0 \quad (2)$$

У цих рівняннях \vec{u} - швидкість рідини. Змінна часу - t , щільність рідини представлена ρ . Тиск всередині рідини (в силових одиницях на одиницю площі) представлений p . Сили тіла (зазвичай просто гравітація) представлені \vec{F} . Нарешті, ν - коефіцієнт рідини кінематична в'язкість [4].

Для прозорих рідин, як вода, в'язкість зазвичай не грає велику роль у зовнішньому вигляді анімації. Якщо вилучити значення в'язкості з нестисливого рівняння Нав'є-Стокса, то отримаємо такий набір рівнянь:

$$\frac{\partial \vec{u}}{\partial t} + \frac{1}{\rho} \nabla p = \vec{F} \quad (3)$$

$$\nabla \cdot \vec{u} = 0 \quad (4)$$

Ці рівняння простіші, і вони називаються рівняннями Ейлера. У фізиці, рівняння Нав'є-Стокса, названого в честь Нав'є і Джорджа Габріеля Стокса, що описують рух в'язкої рідини речовин.

Ці рівняння виникають через застосування другого закону Ісаака Ньютона до руху рідини разом з припущенням, що сила в рідині є сумою

розсіювання (пропорційного градієнту швидкості) і тиску - звідси і опис вузького потоку [7].

Основна відмінність між ними і більш простими рівняннями Ейлера для невязкого потоку полягає в тому, що в рівнянні Нав'є - Стокса також враховується межа Фруда (без зовнішнього поля) і не є рівняннями збереження, а скоріше дисипативної системою, в тому сенсі, що їх не можна привести в однорідну форму (рис. 2.9.).

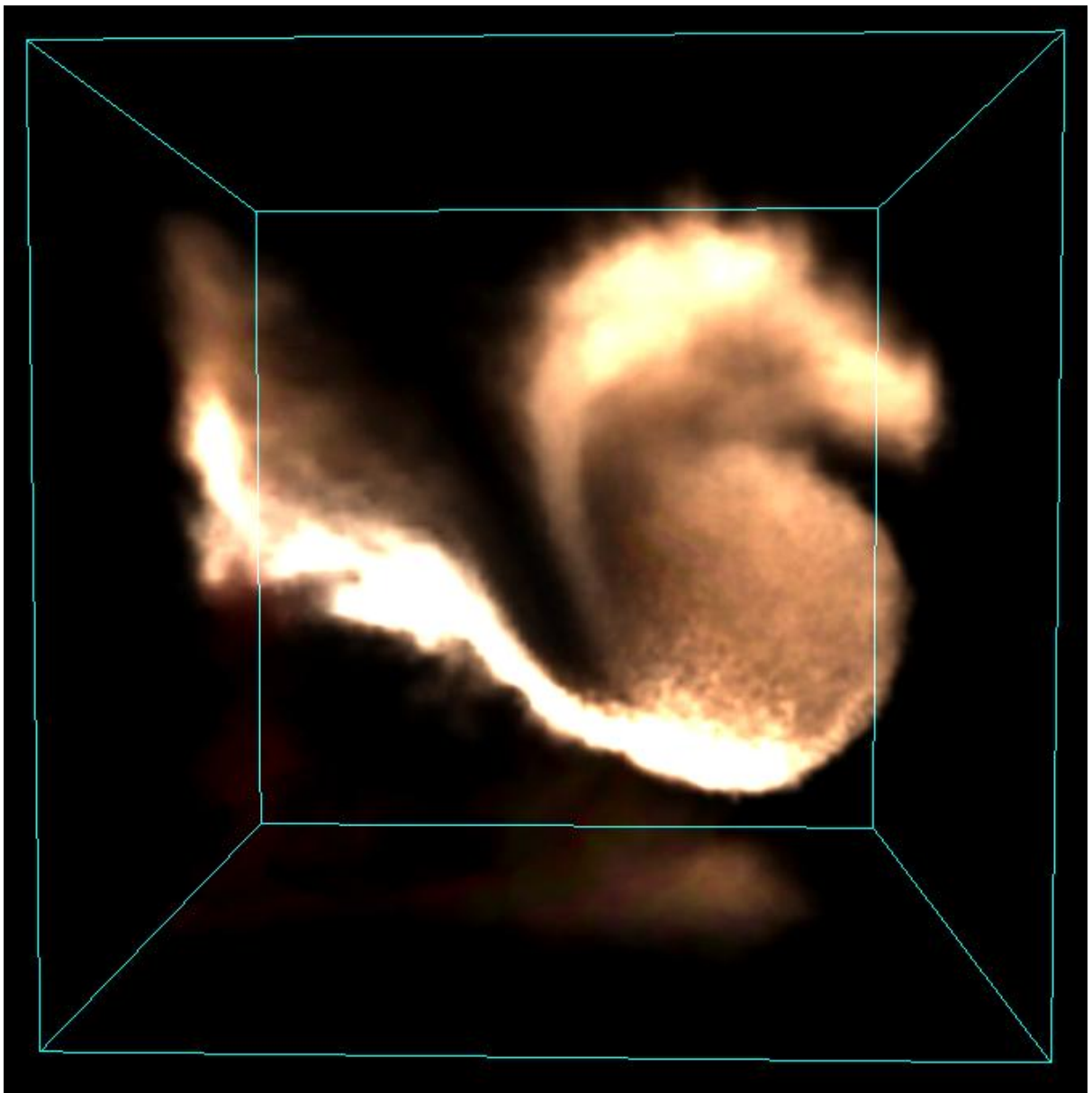


Рис. 2.9. Симуляція потоку на основі тривимірної сітки

Однак потрібен певний тип позначень, щоб описати, який елемент сітки мається на увазі. Для цього використовуються позначення, такі як: $\sim u$ а, b, c для позначення вектора в a, b, c. Нереактивна речовина, яка вводиться в рідину, буде введена нею при одночасному дифундуванні. Загальні приклади цього явища включають моделі, створені молоком, перемішаним у каві, або димом, що піднімається від сигарети.

Симулюючи рідини, стоїть задача моделювати якнайшвидше не втрачаючи числової точності. Тому треба вибрати такий часовий крок, який є максимально великим, але недостатньо великим, щоб дестабілізувати моделювання [21].

Фізичні властивості рідини залежать від її в'язкості. Змінюючи в'язкість, аніматор може імітувати широкий спектр речовин, починаючи від клейової речовини до сильно турбулентних потоків.

Властивості речовини моделюють за допомогою постійної дифузії kS та швидкості дисипації aS . Поряд з цими параметрами, аніматор також повинен вказати значення цих полів на межі сітки.

В основному це два типи: періодичний або фіксований. Граничні умови можуть бути різного типу для кожної координати. Коли вибираються періодичні граничні умови, рідина загортається [20].

Це означає, що частина рідини, яка залишає сітку з одного боку, відновлює сітку з протилежного боку.

У випадку фіксованих меж значення кожної фізичної величини необхідно вказати на межі сітки. Найпростіший метод - встановити значення поля на нуль.

В цілому, підходи на основі частинок менш точні, ніж їх аналоги на основі сітки. Це пов'язано, перш за все, з труднощами роботи з просторовими похідними на неструктурованій хмарі частинок (рис. 2.10.). Проте, симуляції на основі частинок, як правило, набагато простіше програмувати і розуміти.

Крім того, методи на основі частинок набагато швидше і можуть використовуватися в додатках реального часу, таких як відеоігри.

Певні властивості частинок, такі як маса, мають початкові значення на початку моделювання і не повинні змінюватися.

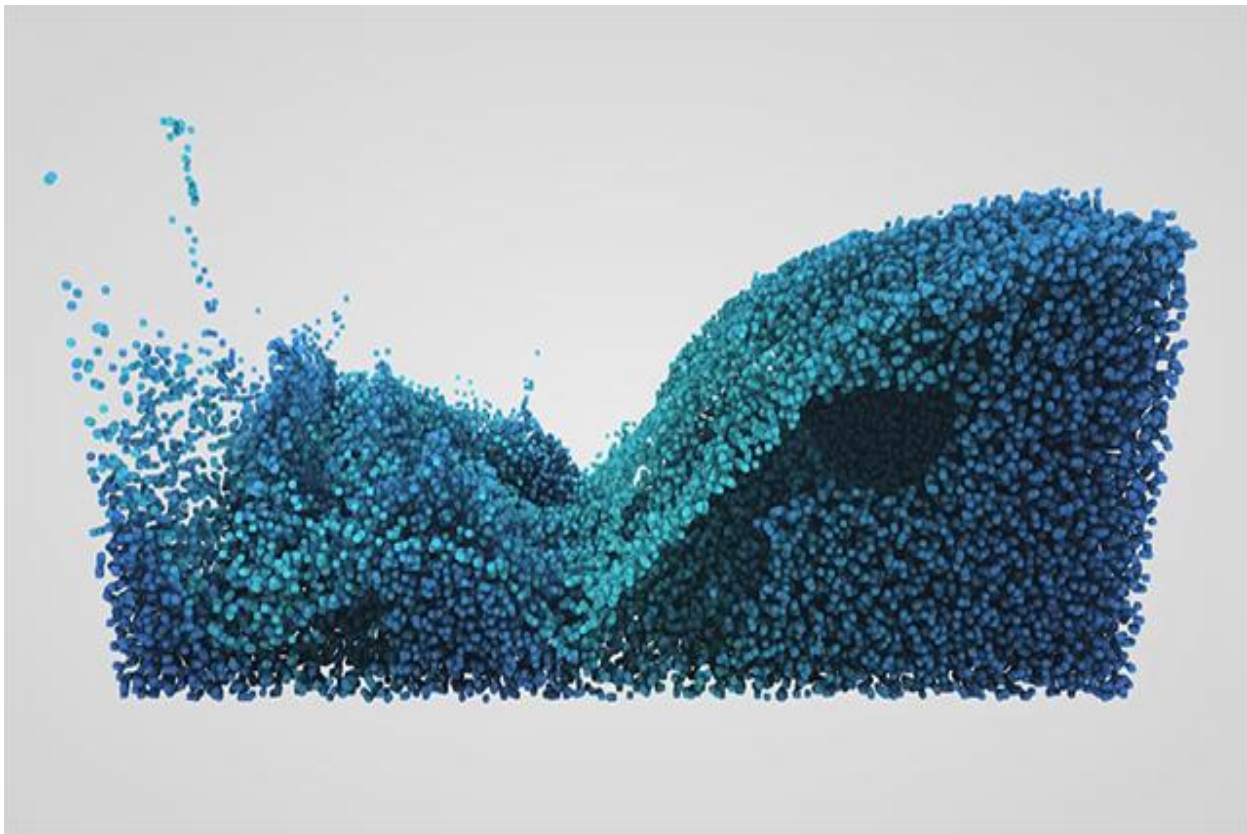


Рис. 2.10. Симуляція потоку на основі частинок

Однак інші властивості повинні перераховуватися на кожному етапі. Розглянемо властивість тиску. Ми маємо дискретну хмару частинок, тому треба використовувати дискретне підсумовування для апроксимації цього інтеграла. Це призводить до рівняння:

$$\sum_{j \neq i}^n M_j W_{R_{ij}} \quad (5)$$

Тут R_{ij} дорівнює евклідовій відстані між часткою i часткою j . Ця функція $W(d)$ називається функцією ядра. Ця функція приймає один скалярний параметр, який представляє собою відстань між двома частинками, і повертає скалярний значення $\in [0, 1]$.

Як правило, функція ядра відображає частки, які знаходяться далі, до значень, близьких до 0. Це має сенс, оскільки частинки, що знаходяться далеко, не чинитимуть великого впливу на інші частинки [24].

Коли частинки знаходяться досить далеко від джерела, функція ядра падає до 0, і, отже, ці частинки більше не потрібно розглядати. Питання про те, яка функція ядра є найкращою, все ще залишається відкритим для досліджень.

Позиційна динаміка.

Найпопулярніші підходи до моделювання динамічних систем у комп'ютерній графіці - засновані на силі.

Накопичувані внутрішні та зовнішні сили, з яких обчислюються прискорення на основі другого закону руху Ньютона. Тоді використовується метод інтеграції часу для оновлення швидкостей і кінцевої позицій об'єкта.

Кілька методів моделювання більшості твердих тіл використовують імпульсну динаміку та безпосередньо маніпулюють швидкістю.

Основна перевага позиційного підходу - це його керованість. Проблеми усунення явних схем інтеграції в діючих системах можна уникнути. Крім того, обмеження зіткнення можна легко обійти, а проникнення можна повністю усунути, виставивши точки на реальні позиції.

Дослідження в галузі фізично коректної анімації в комп'ютерній графіці пов'язані з пошуком нових методів моделювання фізичних явищ, таких як динаміка твердих тіл, деформування об'єктів або потоків рідини.

На відміну від обчислювальних наук, де головна увага приділяється точності, тут основними питаннями є стабільність, надійність та швидкість, а результати повинні залишатися візуально правдоподібними (рис. 2.11.).



Рис. 2.11. Симуляція тканини з використанням позиційної динаміки

Фактично, основною ціллю для проведення досліджень з фізичного моделювання в комп'ютерній графіці є створення спеціалізованих методів, адаптованих до конкретних потреб в даній області [25].

Традиційний підхід до моделювання динамічних об'єктів полягає в роботі з силами. На початку кожного часового кроку накопичуються внутрішні і зовнішні сили. Прикладами внутрішніх сил є сили пружності в деформованих об'єктах або сили в'язкості і тиску в рідинах. Сили тяжіння і зіткнення є прикладами зовнішніх сил.

Другий закон руху Ньютона зв'язує сили з прискореннями через масу. Таким чином, використовуючи щільність або зосереджені маси вершин, сили перетворюються в прискорення. Потім можна використовувати будь-яку схему інтегрування за часом, щоб спочатку обчислити швидкості за прискоренням, а потім позиції за швидкостями.

Деякі підходи використовують імпульси замість сил для управління анімацією. Оскільки імпульси безпосередньо змінюють швидкості, один рівень інтеграції може бути пропущений.

У комп'ютерній графіці і особливо в комп'ютерних іграх часто бажано мати прямий контроль над становищем об'єктів або вершин сітки. Користувач може захотіти прикріпити вершину до кінематичному об'єкту або переконатися, що вершина завжди залишається поза об'єктом, з яким може зіштовхнутися.

Пропонований метод працює безпосередньо з позиціями, що робить такі маніпуляції легшими.

Крім того, при позиційному підході можливо безпосереднє управління інтеграцією, що дозволяє уникнути перерегулювання і проблем з отриманням енергії в зв'язку з інтеграцією.

Висновки до розділу 2

Програмне забезпечення Houdini представляє собою пакет процедурного моделювання, анімації, ефектів, фізичних симуляцій, рендеру та композитингу.

Для роботи з цим програмним забезпеченням, треба розуміти основні принципи процедурного моделювання, тобто ряду методів в комп'ютерній графіці для створення тривимірних моделей і текстур. Процедурне моделювання часто застосовується тоді, коли необхідно створити тривимірні моделі складної форми або коли потрібні більш спеціалізовані інструменти, наприклад моделювання рослин, архітектури, тощо.

Houdini - це процедурний пакет роботи з графікою. Під цей підхід реалізований весь інтерфейс, і логіка будується саме з опорою на процедурність.

За допомогою функцій, які надає Houdini можна виконати побудову моделей, об'єктів, симуляцій та їх налаштування, а також перевірити їх роботу запускаючи та тестуючи симуляції безпосередньо в редакторі.

Було розглянуто особливості роботи з симуляціями потоків, таких як різні види рідин, вогонь та інші.

РОЗДІЛ 3

ПРОГРАМНА РЕАЛІЗАЦІЯ

3.1. Створення 3D симуляції засобами Houdini

Розглянемо на прикладі створення симуляції потоків рідини робочій процес Houdini. Для початку роботи з моделюванням потоку рідини треба створити об'єкт, який буде джерелом потоку. У даному випадку це буде примітив у вигляді сфери (рис. 3.1.).

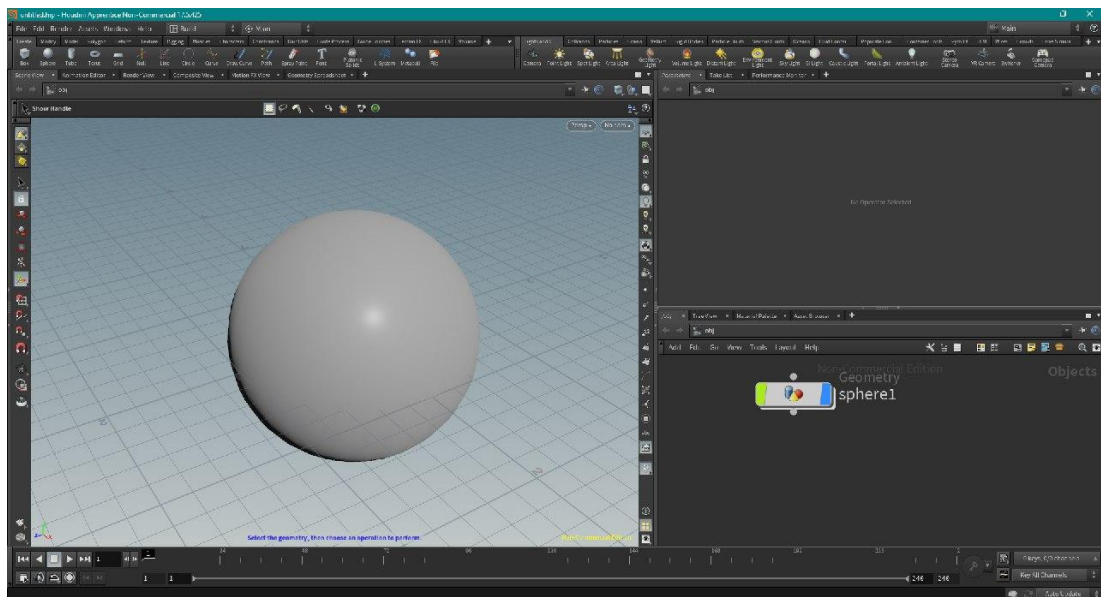


Рис. 3.1. Примітив сфера

Після цього додаємо ноду transform, яка дозволить змінювати такі параметри сфери як розмір, позицію та поворот (рис. 3.2.).

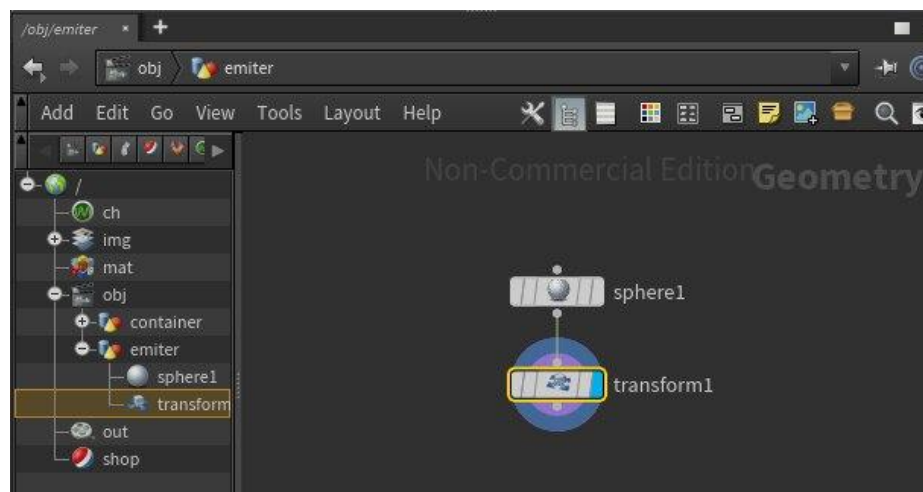


Рис. 3.2. Нода transform

Для того, щоб візуалізувати, як рідина з різною в'язкістю буде стікати по поверхні, треба додати ще один об'єкт, який буде цю поверхню імітувати (рис. 3.3.).

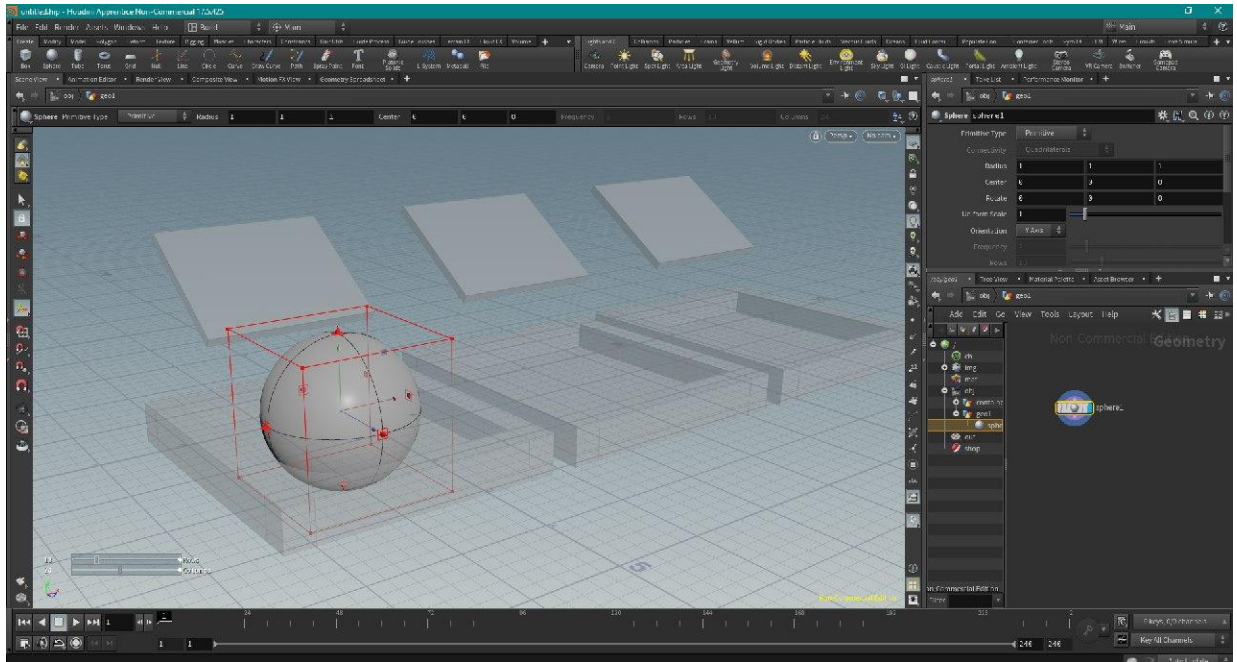


Рис. 3.3. Об'єкт з похилою поверхнею та контейнером

Змінимо розмір та позицію сфери за допомогою ноди transform так, щоб вона знаходилась над похилою поверхнею (рис. 3.4.).

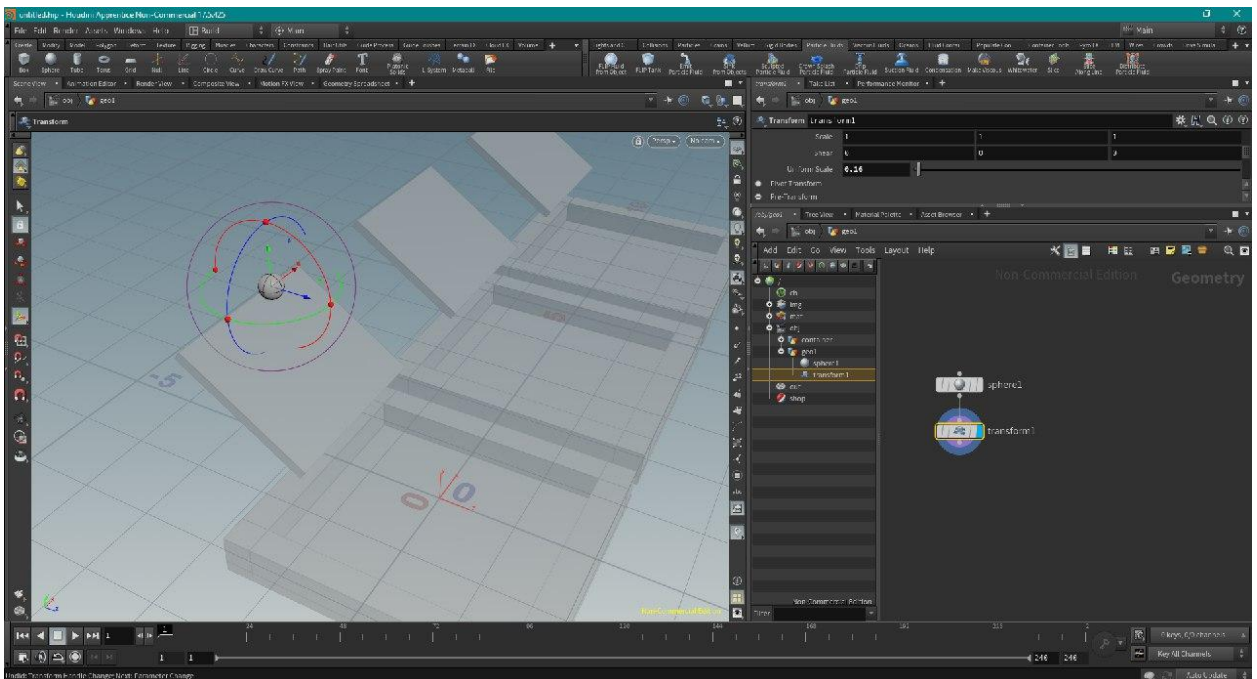


Рис. 3.4. Розміщення сфери над поверхнею

Використаємо ноди `flipsource` та `DOP Network` для того, щоб додати сфері властивості джерела частинок, які будуть імітувати потік рідини (рис. 3.5.).

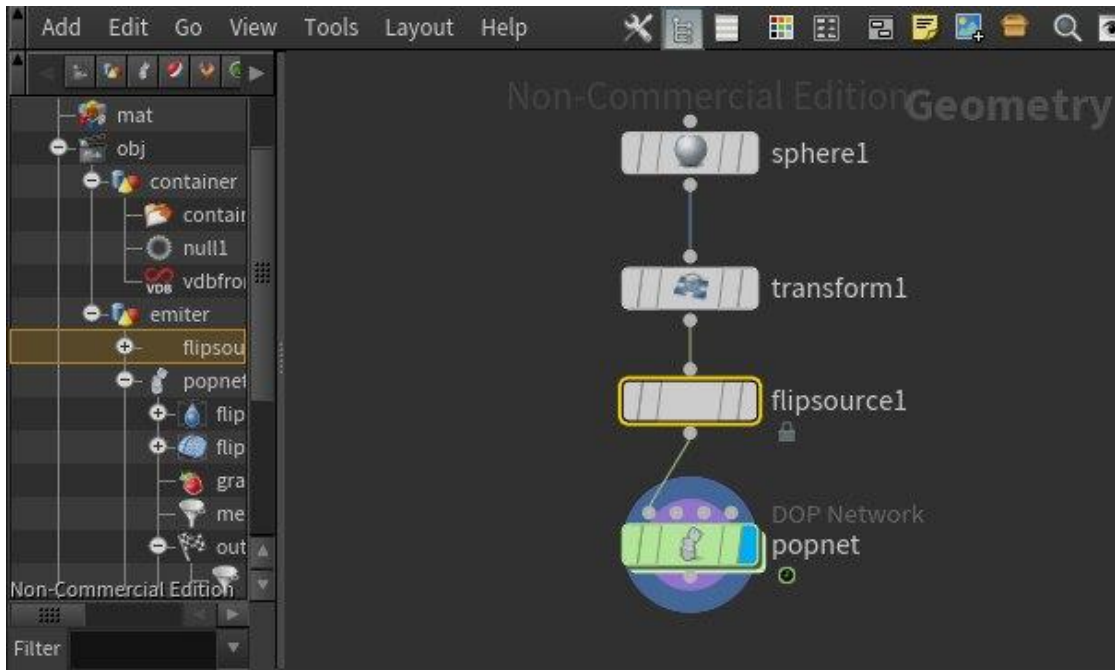


Рис. 3.5. Ноди `flipsource` та `DOP Network`

В свою чергу, нода `DOP Network` містить в собі ще одну мережу нодів `flipsolver`, які відповідають саме за створення частинок. (рис. 3.6.)

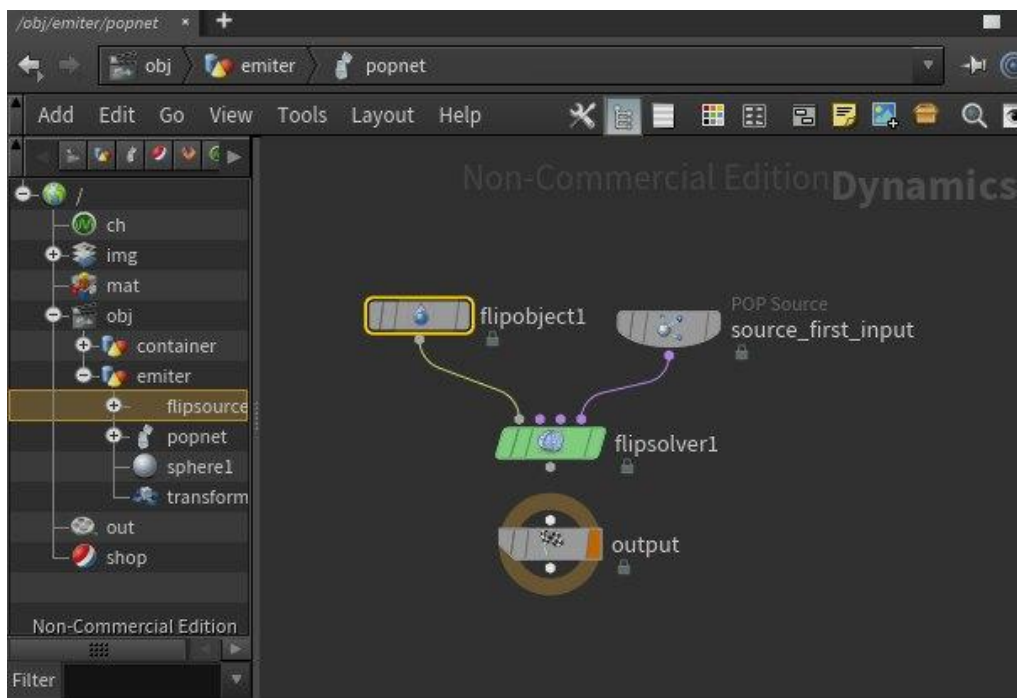


Рис. 3.6. Мережа `flipsolver`

Під'єднавши мережу flpsolver до сфери, можна побачити, що навколо неї сформувалась хмара частинок (рис. 3.7.).

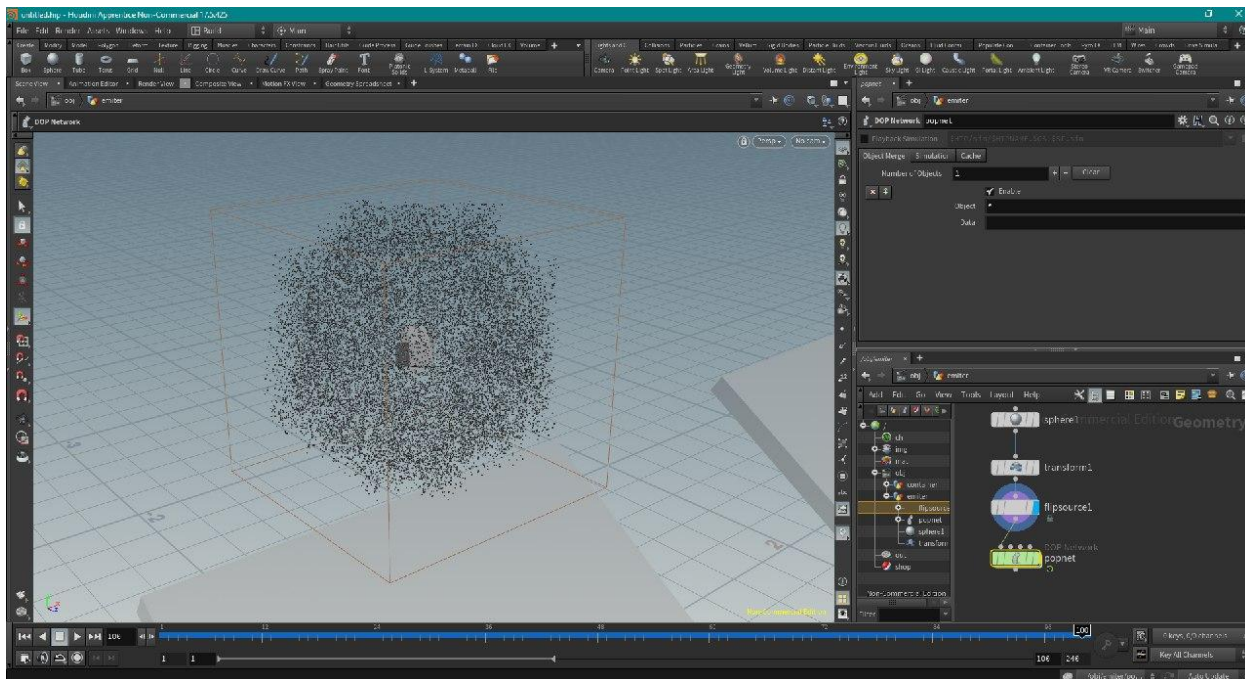


Рис. 3.7. Результат дії мережі flpsolver на сферу

Зараз частинки лише створюються навколо сфери. Для того, щоб імітувати поведінку потоку рідини, треба підключити до мережі flpsolver ноду gravity. Вона змусить частинки падати вертикально вниз (рис. 3.8.).

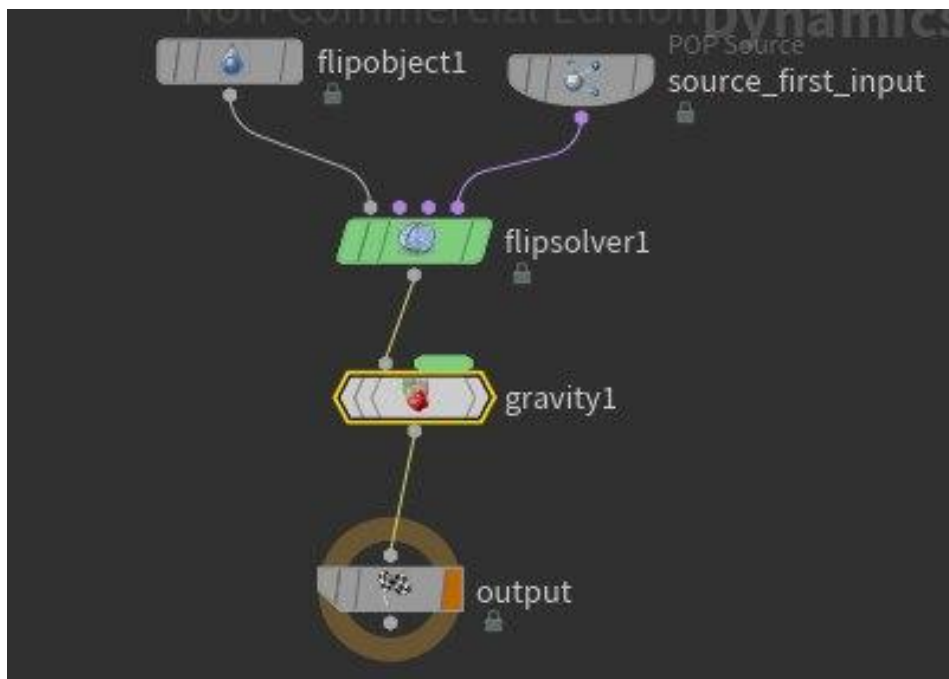


Рис. 3.8. Нода gravity

Увімкнувши симуляцію, можна побачити, що поведінка частинок стала більше схожа на поведінку потоку рідини (рис. 3.9.).

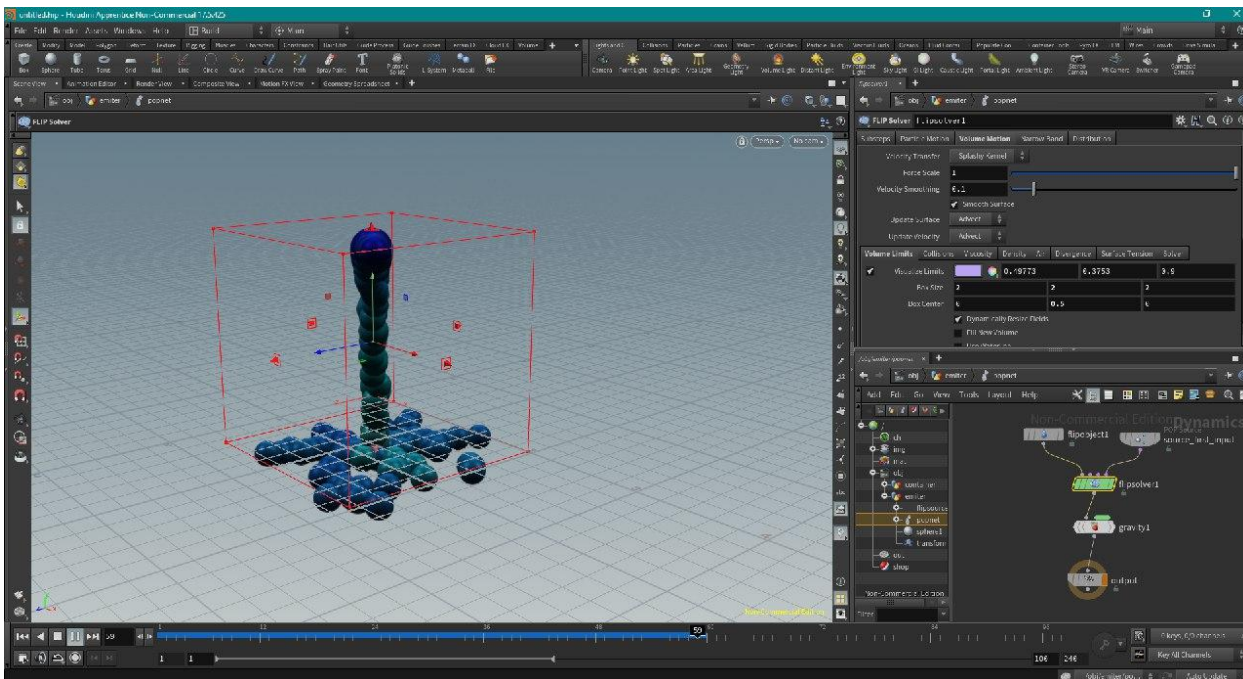


Рис. 3.9. Результат дії ноди gravity

Але наразі, частинки лише падають вниз у межах заданої області та ніяк не взаємодіють з поверхнею, яка була додана раніше. Щоб це реалізувати, потрібно додати посилання на статичний об'єкт до мережі flipsolver, а саме ноди staticobject та staticsolver (рис. 3.10.).

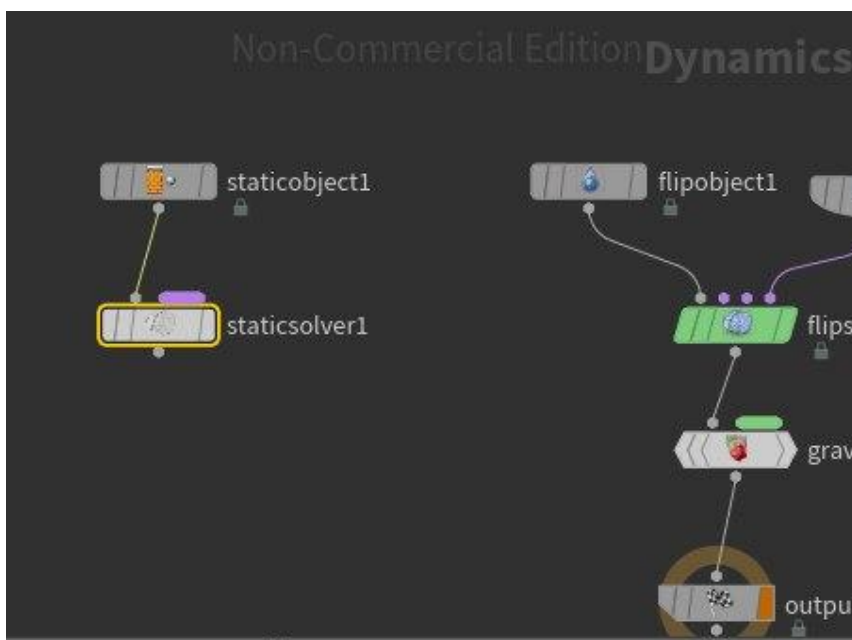


Рис. 3.10. Ноди staticobject та staticsolver

Виконавши попередні маніпуляції та увімкнувши симуляцію, можна побачити, що частинки фізично коректно падають вертикально вниз та стікають з похилої поверхні до контейнеру (рис. 3.11.).

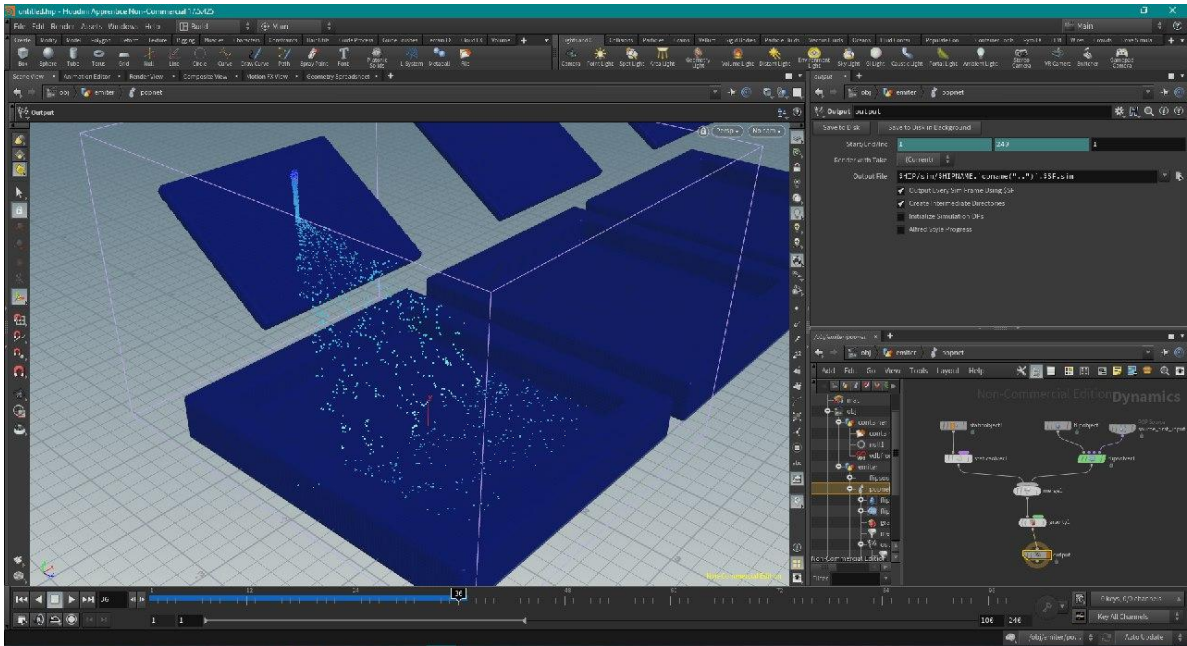


Рис. 3.11. Симуляція потоку частинок

На даному етапі, потік частинок поводить себе як рідина, яка не має таких властивостей як в'язкість, температура або густина. Ці властивості можна додати у вікні параметрів мережі `flipsolver`. За в'язкість рідини відповідає параметр `Viscosity`. Надаємо йому значення 100 одиниць (рис. 3.12.).

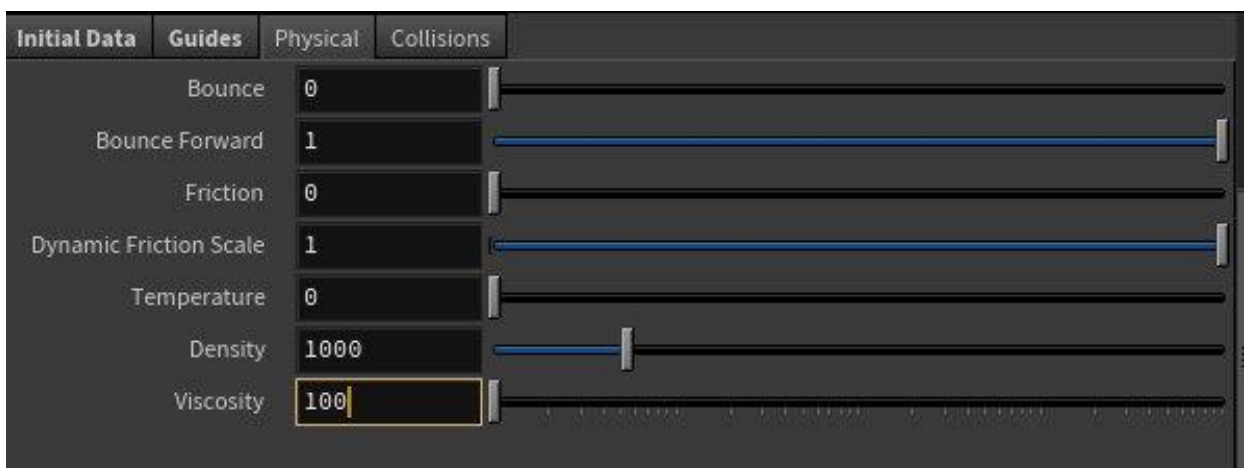


Рис. 3.12. Вікно налаштування фізичних параметрів рідини

Знову запустивши симуляцію, можна побачити, як параметр Viscosity діє на поведінку потоку рідини. Потік стікає з поверхні набагато повільніше та має вигляд в'язкої субстанції, такої як мед (рис. 3.13.).

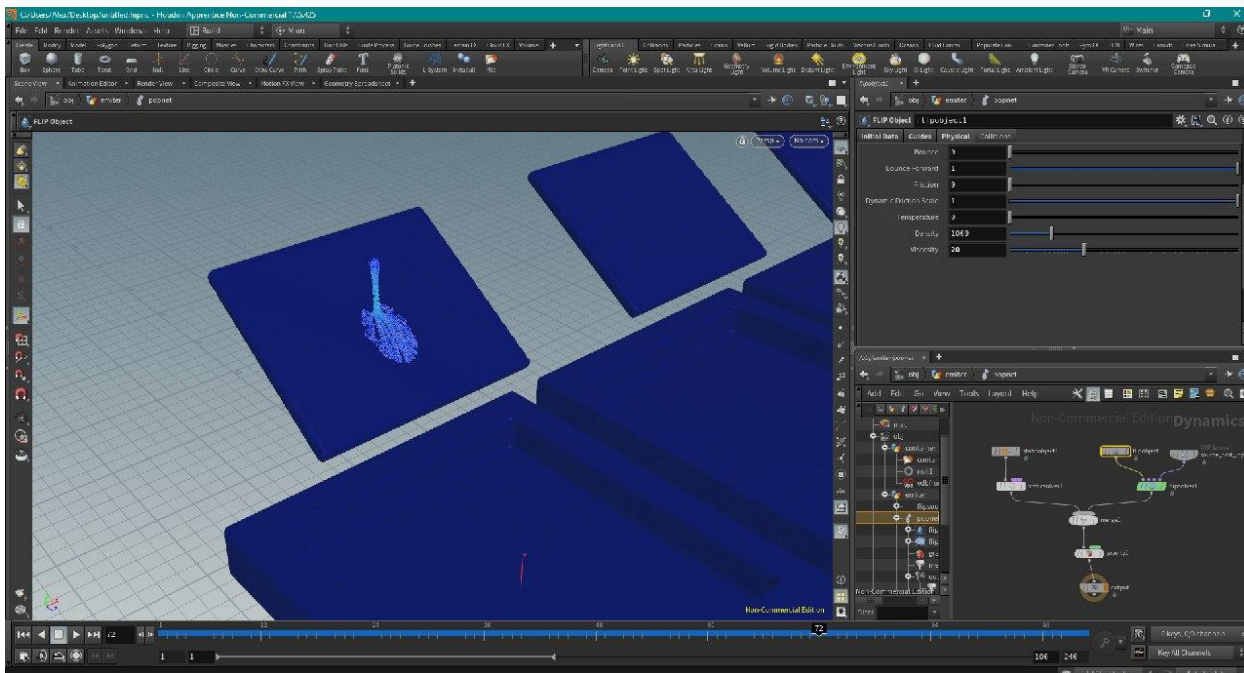


Рис. 3.13. Симуляція потоку частинок з параметром Viscosity

Результат поведінки рідин з різною в'язкістю можна побачити на (рис. 3.14.).

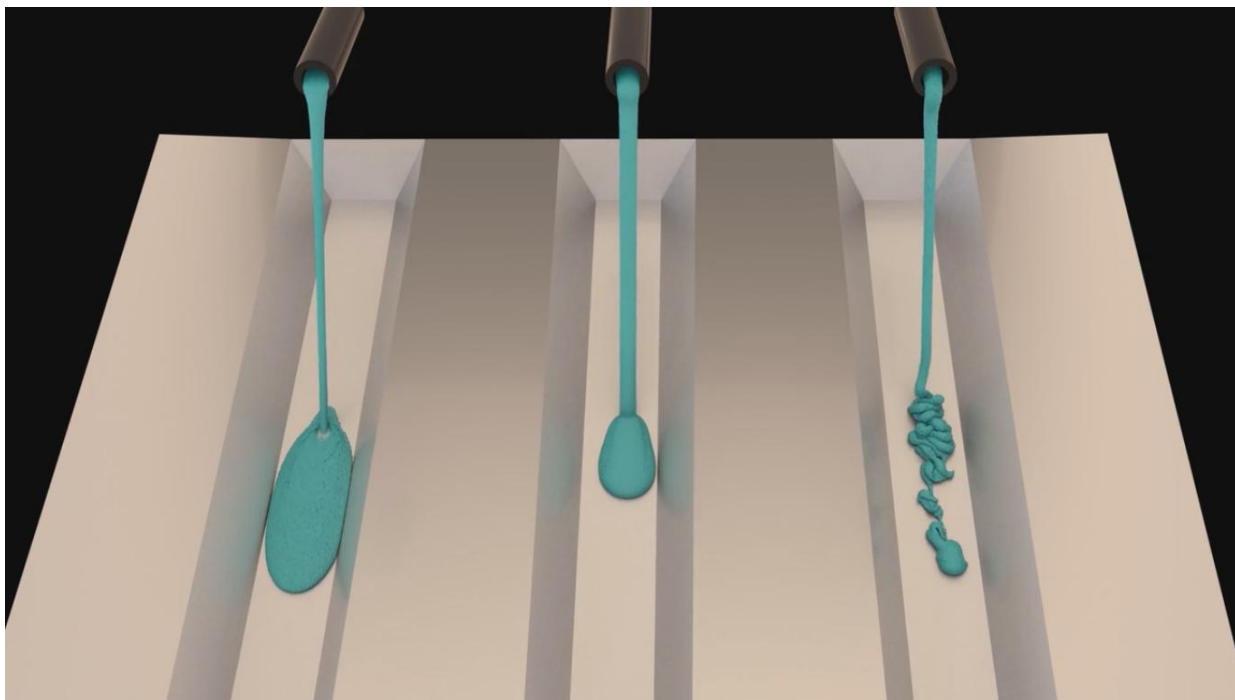


Рис. 3.14. Симуляція рідин з різною в'язкістю

3.2. Тестування та аналіз проблем

У ході роботи з симуляцією потоків рідин виникли такі проблеми:

Якщо для частинок з низькою роздільною здатністю встановлено значення «Балістична обробка», мережа FLIP solver виявить частки, що залишають поверхню і є занадто маленькими, щоб розчинитися в сітці, і обробить їх балістичним методом.

Це добре працює для рідини без в'язкості, але для в'язких рідин це може привести до сповзання частинок з поверхні навіть при дуже високих значеннях в'язкості.

Якщо для методу Under-Resolved Particles задано значення No Detection, це поведінка відключається і має припинити повзучість частинок.

При спливанні в'язких часток які дуже повільно рухаються іноді можуть бути видні частки, недавно введені в результаті процесу пересіву. Відключення повторного посіву має полегшити цю проблему [24].

Вибір 32-розрядного числа з плаваючою комою для точності на вкладці «В'язкість» прискорить визначення в'язкості і буде достатньою точністю для більшості симуляцій. Якщо у вас швидкий графічний процесор, включення прискорення OpenCL за допомогою параметра Use OpenCL на вкладці Solver може ще більше прискорити моделювання в'язких рідин з високою роздільною здатністю (рис. 3.15.).

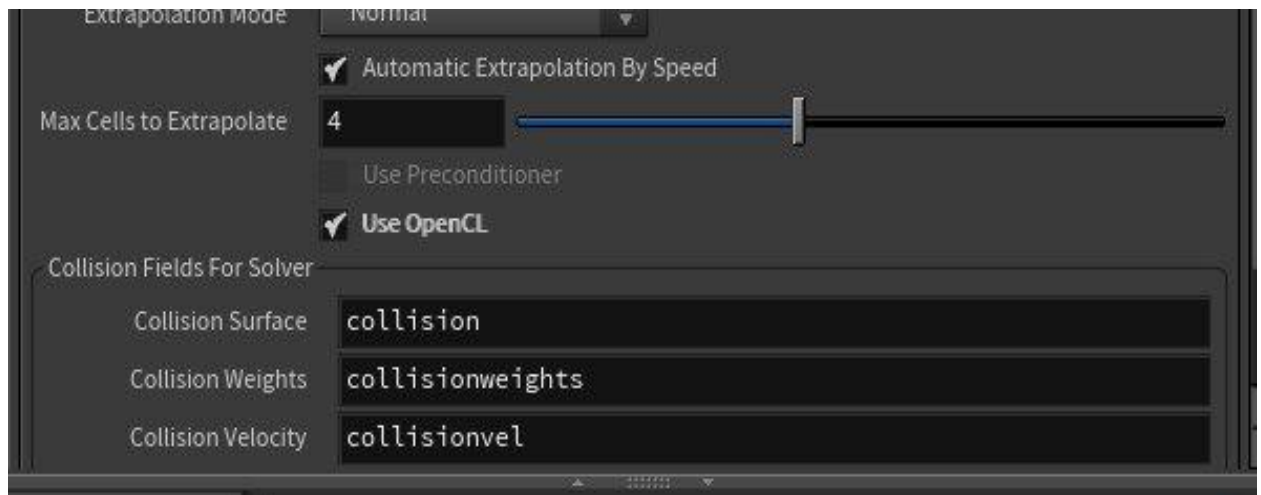


Рис. 3.15. Вибір параметру Use OpenCL на вкладці Solver

Під час додавання ноди flpsolver до примітива сфери, можна побачити, що створювані частинки мають вигляд сфер (рис. 3.16.). Для коректної симуляції потоків рідини, частинки мають бути набагато меншого розміру.

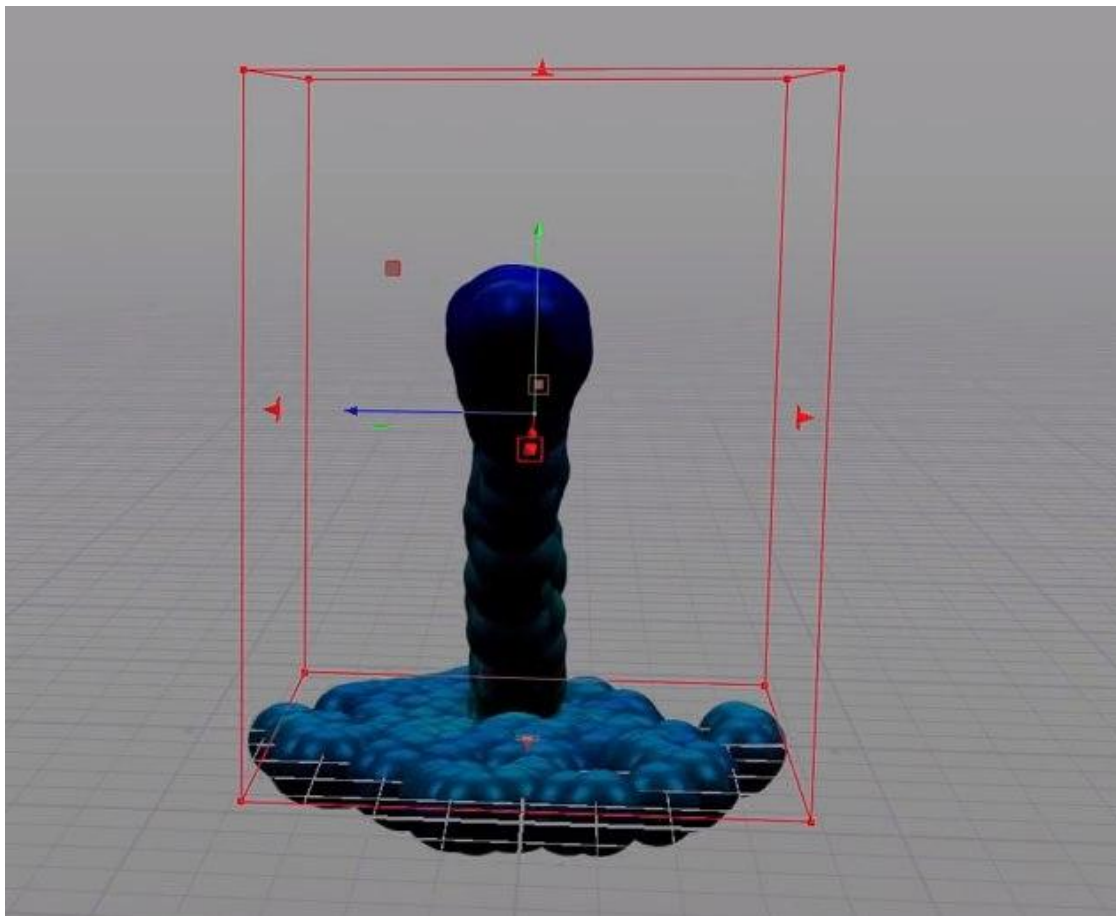


Рис. 3.16. Відображення частинок у вигляді сфер

Для вирішення цієї проблеми треба переключити режим відображення частинок на particles(рис. 3.17.).

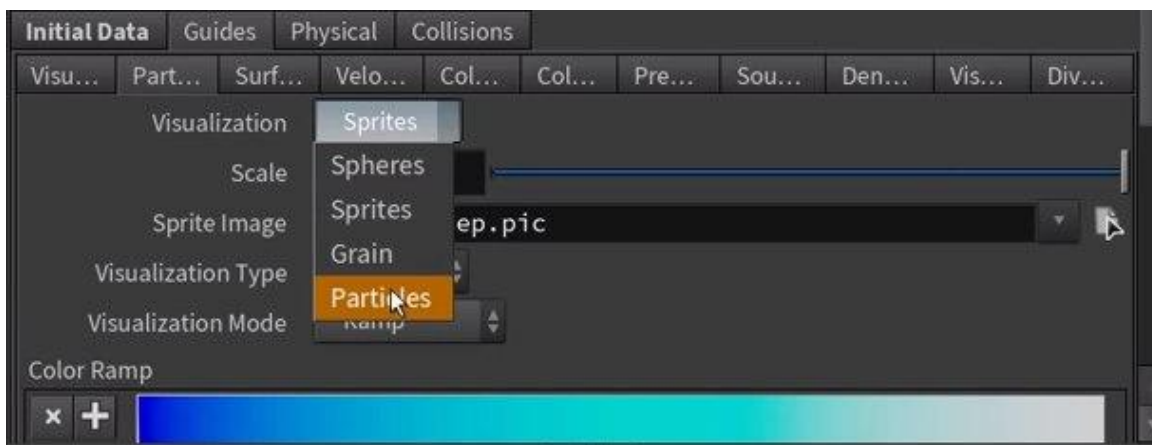


Рис. 3.17. Меню вибору методу відображення частинок

Після додавання об'єкту похилої площини з контейнером, виникла проблема, коли частинки під час симуляції не стікали, а проходили крізь неї (рис. 3.18.).

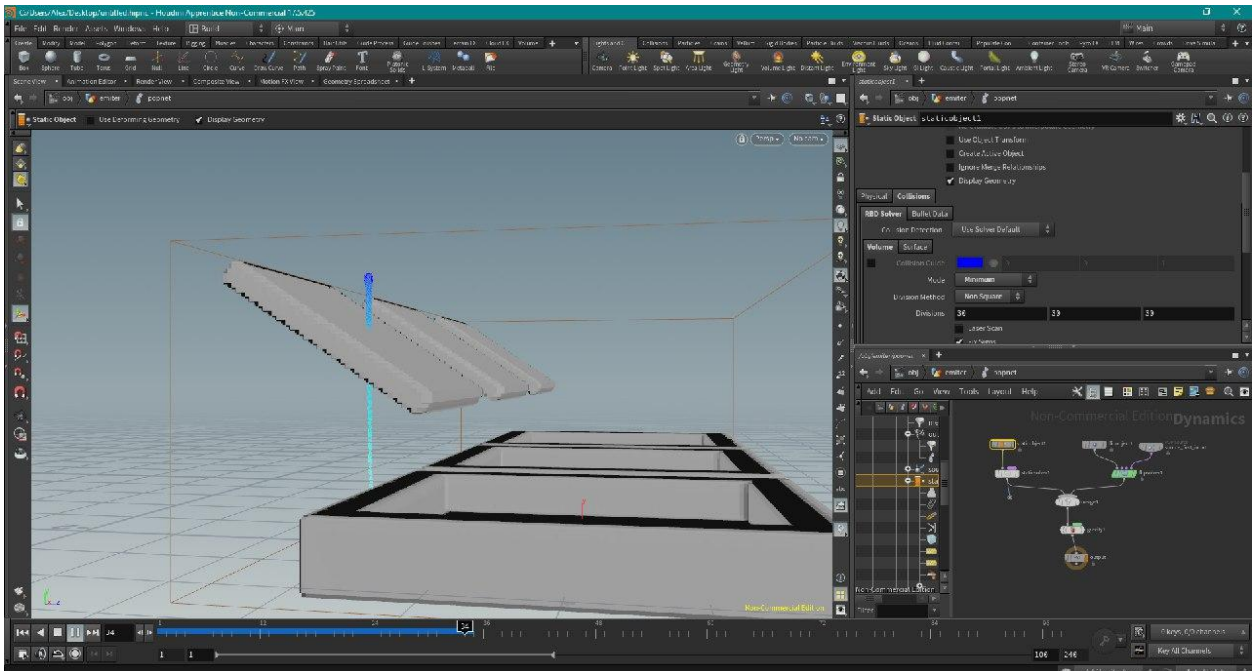


Рис. 3.18. Ігнорування частинками поверхні для зіткнення

Для того, щоб фізична модель поверхні працювала коректно, треба переключити модель зіткнення на Volume Sample (рис. 3.19.).

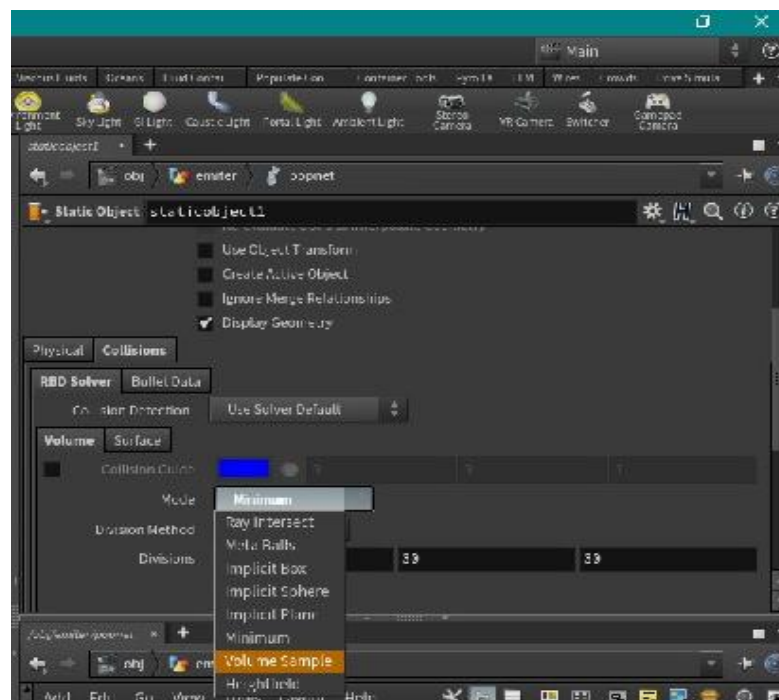


Рис. 3.19. Меню вибору моделі зіткнення

Якщо стінка контейнера занадто тонка, частинки можуть зіткнутися зі стінкою, але етап визначення тиску може не побачити її, змушуючи частки «складатися» в стінку замість того, щоб відскакувати від неї, створюючи враження, що рідина стискається і зникає.

Збільшення товщини геометрії зіткнення виправить це (також можна використовувати іншу геометрію для створення поля зіткнення і для рендеринга, так що є можливість візуалізувати тонку стінку контейнера).

Якщо ви бачите несподівану поведінку при зіткненні, включіть опцію візуалізації Collision Velocity на об'єкті FLIP і переконайтеся, що solver визначає значення швидкості.

Щоб створити більш інтенсивний потік рідини з джерела, можна використовувати різноманітні форми об'єктів (наприклад, тор або трубу, а не сферу) і додати шум до швидкостей випромінювання (рис. 3.20.).

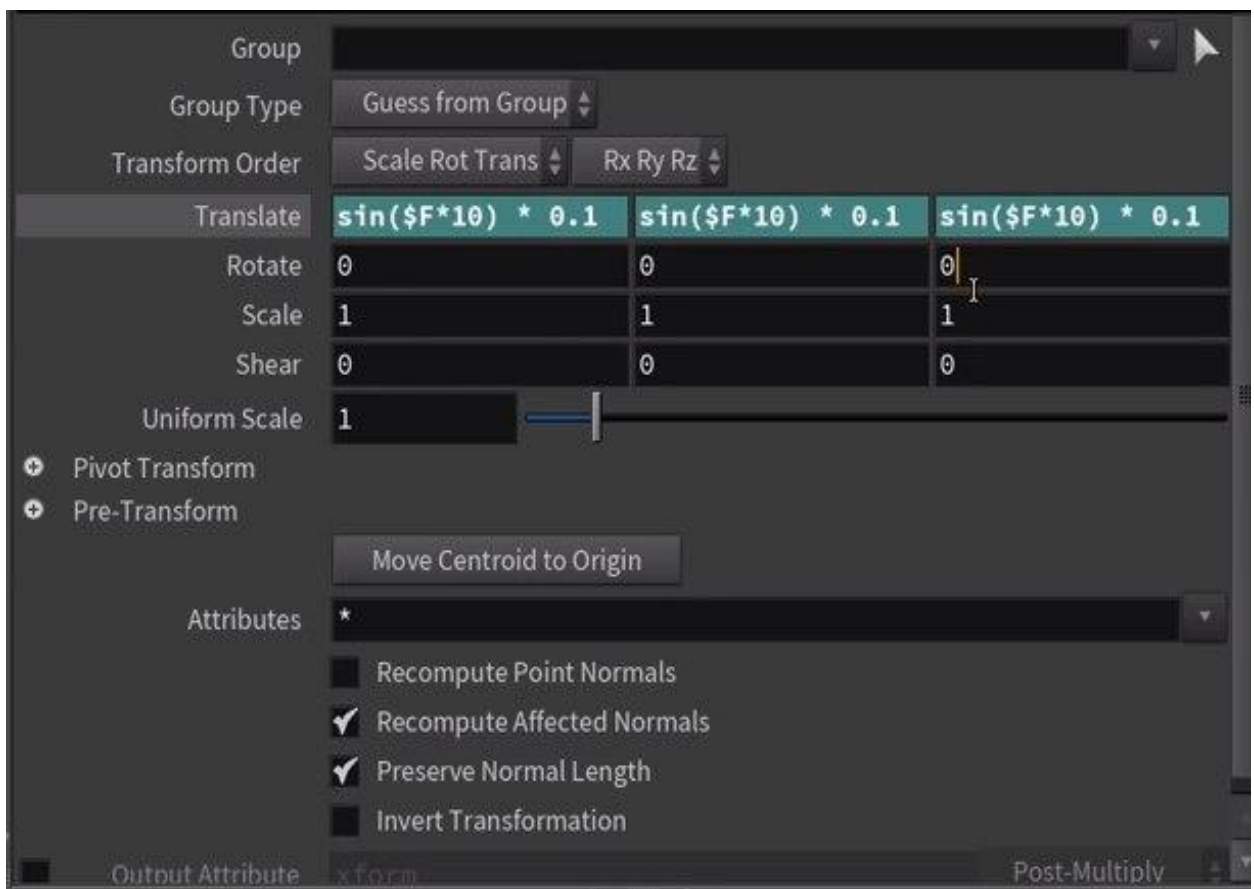


Рис. 3.20. Зміна значень швидкостей випромінювання

Частинки рідини використовують значення атрибута v (швидкість) в точці, з якої вони народжуються, як їх початкової швидкості, тому ви можете контролювати швидкості випромінювання, задавши v в геометричній мережі вихідного об'єкта (рис. 3.21.).

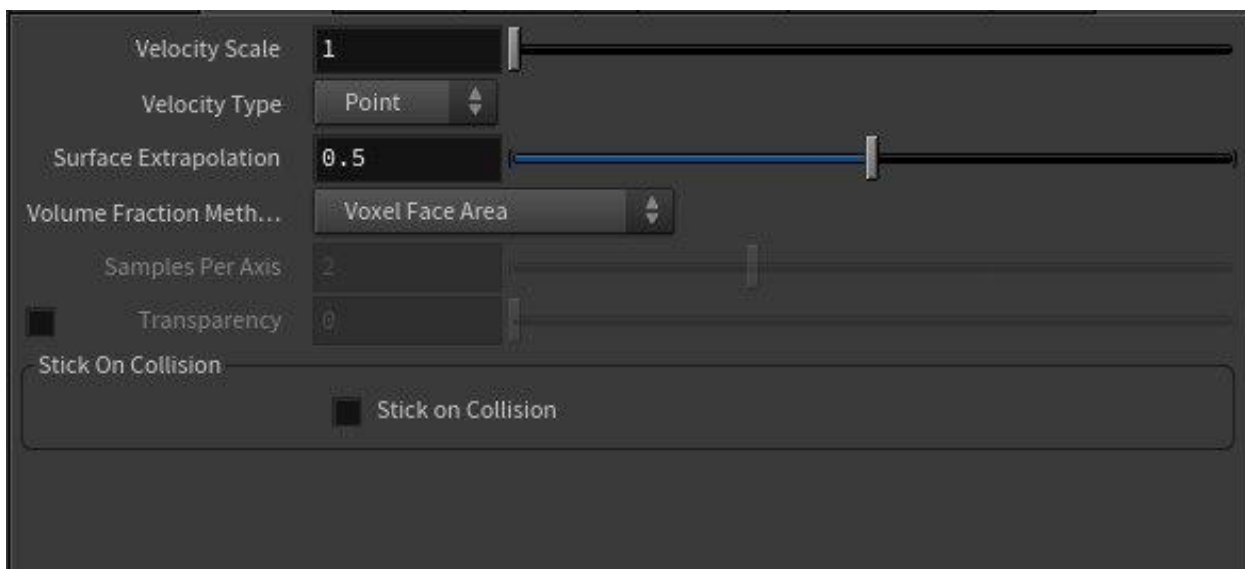


Рис. 3.21. Параметр velocity

Висновки до розділу 3

За допомогою програмного забезпечення Houdini було створено симуляцію потоків рідин на основі системи часток, реалізовано такі базові елементи симуляції як:

- Система створення частинок на основі джерела у вигляді сфери;
- Взаємодія частинок з іншими об'єктами сцени
- Реалізовані потоки мали різні фізично коректні значення

в'язкості

Було розглянуто основні проблеми, які з'являлися під час роботи над моделюванням симуляції та методи їх вирішення.

ВИСНОВКИ

В ході аналізу з'ясовано переваги візуального сприйняття для взаємодії з людьми, так як інформація у вигляді зображень засвоюється швидше, ніж у текстовому вигляді та краще запам'ятовується. З розвитком технологій з'явилася така область комп'ютерних наук, як комп'ютерна графіка, яка вивчає методи цифрового синтезу і маніпулювання візуальним контентом. Було виявлено сфери застосування комп'ютерної графіки, а саме в навчанні, медицині, в кіновиробництві, створення комп'ютерних ігор, в архітектурі, тощо. Також були розглянуті основні підходи до створення 3D-графіки та виконано порівняння існуючого спеціалізованого програмного забезпечення.

Для створення анімацій, ефектів, симуляції, тощо, використовується процедурне моделювання - це загальний термін для позначення ряду методів в комп'ютерній графіці для створення тривимірних моделей і текстур з наборів правил. Було обрано програмне забезпечення Houdini, яке являється індустріальним стандартом у створенні візуальних фізичних ефектів, наприклад симуляції вибуху, руйнування твердих тіл, рух рідини, волосся або хутра. Було розглянуто використання фізичних законів класичної механіки у комп'ютерній графіці, а саме загальне рівняння нестисливих рідин Нав'є Стокса, другого закону руху Ньютона.

Було розглянуто принципи створення 3D-анімаційних презентацій на прикладі симуляції потоку рідини з різними значеннями в'язкості за допомогою системи часток програмного забезпечення Houdini. Розглянуто основні проблеми, які виникають під час створення симуляцій та методи їх рішення. Проведено тестування створеного проекту, виявлено та усунуто помилки, оцінено недоліки та шляхи удосконалення його у майбутньому.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. 3D modeling and rendering software for design visualization, games, and animation [Електронний ресурс] – Режим доступу: <https://www.autodesk.com/products/3ds-max/overview> Назва з екрана.
2. Інтерфейс Houdini і логіка [Електронний ресурс] – Режим доступу: <https://kondratiki.pro/core/kurs-houdini-uroki/interfeis-houdini> Назва з екрана.
3. Романюк А. Н. РОЛЬ КОМП'ЮТЕРНОЇ ГРАФІКИ В ОБРАЗОВАННІ [Електронний ресурс] Режим доступу: http://elar.urfu.ru/bitstream/10995/26617/1/notv_2013_161.pdf Назва з екрана.
4. Уравнение Навье – Стокса [Електронний ресурс] – Режим доступу: <http://www.claymath.org/millennium-problems/navier%E2%80%93stokes-equation> Назва з екрана.
5. Blender 2.8: EEVEE [Електронний ресурс] – Режим доступу: <https://blender3d.com.ua/blender-2-8-eevee/> Назва з екрана.
6. Blender Developers Blog [Електронний ресурс] – Режим доступу: <https://code.blender.org/> Назва з екрана.
7. Braley C. Fluid Simulation For Computer Graphics: A Tutorial in Grid Based and Particle Based Methods [Електронний ресурс] / C. Braley, A. Sandu – Режим доступу: <http://users.encs.concordia.ca/~grogono/Graphics/fluid-5.pdf> Назва з екрана.
8. Clustering smoke and pyro simulations [Електронний ресурс] – Режим доступу: <https://www.sidefx.com/docs/houdini17.5/pyro/clustering.html> Назва з екрана.

9. Computer graphics [Электронный ресурс] – Режим доступа: https://www.newworldencyclopedia.org/entry/Computer_graphics Назва з екрана.
10. Hair and Fur [Электронный ресурс] – Режим доступа: <https://www.sidefx.com/docs/houdini17.5/fur/index.html> Назва з екрана.
11. HOUDINI DOCUMENTATION [Электронный ресурс] – Режим доступа: <https://www.sidefx.com/docs/> Назва з екрана.
12. Introduction to Fluid Simulations in Houdini [Электронный ресурс] – Режим доступа: <https://app.pluralsight.com/course-player?name=introduction-fluid-simulations-houdini-2078-m1&mode=live&clip=0&course=introduction-fluid-simulations-houdini-2078> Назва з екрана.
13. Jacobs T. The Advantages of 3D Design [Электронный ресурс] / Tim Jacobs. – 2014. – Режим доступа: <https://plusgroups.com/the-advantages-of-3d-design/> Назва з екрана.
14. Jarratt S. The best 3D modelling software 2019 [Электронный ресурс] / Steve Jarratt. – 2019. – Режим доступа: <https://www.creativebloq.com/features/best-3d-modelling-software> Назва з екрана.
15. Learning Houdini [Электронный ресурс] – Режим доступа: https://www.sidefx.com/learn/getting_started/ Назва з екрана.
16. Lytle S. Top 10 Benefits of 3D Visualization [Электронный ресурс] / Sam Lytle. – 2016. – Режим доступа: <http://www.civilfx.com/benefits-3d-architectural-visualization/> Назва з екрана.
17. Махон [Электронный ресурс] – Режим доступа: <https://www.maxon.net/ru/produkty/cinema-4d/obzor/> Назва з екрана.
18. Petty J. What is 3D Modeling & What's It Used For? [Электронный ресурс] / Josh Petty – Режим доступа: <https://conceptartempire.com/what-is-3d-modeling/> Назва з екрана.

19. Pixologic [Электронный ресурс] – Режим доступа: <https://pixologic.com/> Назва з екрана.
20. Simulating pressure in a grid based liquid simulation [Электронный ресурс] – Режим доступа: <https://gamedev.stackexchange.com/questions/58734/simulating-pressure-in-a-grid-based-liquid-simulation/58736#58736> Назва з екрана.
21. Stam J. Stable Fluids [Электронный ресурс] / Jos Stam // ACM Transactions on Graphics. – 1999. – Режим доступа: https://www.researchgate.net/publication/2486965_Stable_Fluids Назва з екрана.
22. Takahashi T. Video-guided real-to-virtual parameter transfer for viscous fluids [Электронный ресурс] / Т. Takahashi, М. Lin // ACM Transactions on Graphics. – 1999. – Режим доступа: https://www.researchgate.net/publication/337130101_Video-guided_real-to-virtual_parameter_transfer_for_viscous_fluids Назва з екрана.
23. Viscosity [Электронный ресурс] – Режим доступа: <https://www.sidefx.com/docs/houdini/fluid/flipviscosity.html> Назва з екрана.
24. Viscous fluid tools [Электронный ресурс] – Режим доступа: <https://www.sidefx.com/docs/houdini/fluid/lava.html> Назва з екрана.
25. VOLINO P. Versatile and Efficient Techniques for Simulating Cloth and Other Deformable Objects [Электронный ресурс] / P. VOLINO, M. COURCHESNE, N. THALMANN – Режим доступа: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.89.3207&rep=rep1&type=pdf> Назва з екрана.