

Міністерство освіти і науки України
Національний технічний університет
«Дніпровська політехніка»

Інститут електроенергетики
(інститут)

Факультет інформаційних технологій
(факультет)

Кафедра Програмного забезпечення комп'ютерних систем
(повна назва)

ПОЯСНЮВАЛЬНА ЗАПИСКА
кваліфікаційної роботи ступеня
бакалавра

(назва освітньо-кваліфікаційного рівня)

студента

Савостенка Максима Андрійовича
(ПІБ)

академічної групи

122-20-1
(шифр)

спеціальності

122 Комп'ютерні науки
(код і назва спеціальності)

освітньої програми

Комп'ютерні науки
(назва освітньої програми)

на тему:

*Розробка інженерного плагіну
для програмного комплексу САПР SolidWorks*

Керівники	Прізвище, ініціали	Оцінка за шкалою		Підпис
		рейтинговою	інституційною	
кваліфікаційної роботи	<i>доц. Мартиненко А.А.</i>			
розділів:				
спеціальний	<i>доц. Мартиненко А.А.</i>			
економічний	<i>доц. Касьяненко Л.В.</i>			
Рецензент	<i>доц. Шедловський І.А.</i>			
Нормоконтролер	<i>доц. Мартиненко А.А.</i>			

Дніпро
2024

Міністерство освіти і науки України
НТУ «Дніпровська політехніка»

ЗАТВЕРДЖЕНО:

завідувач кафедри
програмного забезпечення комп'ютерних систем
(повна назва)

_____ М.О. Алексєєв
(підпис) (прізвище, ініціали)

« » _____ 2024 року

ЗАВДАННЯ
на кваліфікаційну роботу
бакалавра
(назва освітньо-кваліфікаційного рівня)

студента 122-20-1 Савостенка Максима Андрійовича
(група) (прізвище та ініціали)

тема кваліфікаційної роботи Розробка інженерного плагіну
для програмного комплексу САПР SolidWorks

затверджена наказом ректора НТУ «ДП» від «23» травня 2024 р. № 469-с

Розділ	Зміст виконання	Термін виконання
Спеціальний	На основі матеріалів проектно-технологічної практики та інших науково-технічних джерел провести аналіз стану рішення проблеми та постановку задачі. Обґрунтувати вибір та здійснити реалізацію методів вирішення проблеми	03.06.2024 р.
Економічний	Провести розрахунок трудомісткості розробки інженерного плагіну для програмного комплексу САПР SolidWorks, витрат на створення програмного забезпечення й тривалості його розробки	10.06.2024 р.

Завдання видав _____ доц. Мартиненко А.А.
(підпис) (посада, прізвище, ініціали)

Завдання прийняв до виконання _____ Савостенко М.А.
(підпис) (прізвище, ініціали)

Дата видачі завдання: 14.01.2024 р.

Термін подання кваліфікаційної роботи до ЕК: 24.06.2024 р.

ВСТУП

Актуальність теми. На сучасному етапі автоматизація процесів проєктування та виробництва набуває все більшої популярності на ринку технологій. Особливо актуальною є автоматизація певних етапів ливарної технології, яка дозволяє значно зменшити витрати часу та ресурсів, а також підвищити точність і якість кінцевої продукції. Використання вбудованого в систему SolidWorks API для автоматизованого проєктування надає можливість оптимізувати процеси, що зменшує необхідність ручної праці та мінімізує людський фактор у розрахунках і проєктуванні формувальних операцій.

Досвідчений інженер, використовуючи професійні інструменти, може здійснювати проєктування елементів ливарної технології вручну, проте навіть професіонал зазнає труднощів при виконанні складних розрахунків та створенні 3D моделей з високою точністю. Автоматизовані функції, інтегровані в SolidWorks через API, дозволяють значно спростити цей процес. Для системи автоматизації типові завдання, такі як розрахунки та формування моделей, можуть бути виконані швидко та ефективно, що є особливо важливим для виробництв з великим обсягом замовлень.

Система автоматизованого проєктування на основі SolidWorks API може бути використана для розробки різноманітних ливарних технологій, включаючи створення форм, моделювання процесів лиття та оптимізацію матеріалів. Це дозволяє значно підвищити продуктивність та якість продукції, що виготовляється. З огляду на це, розробка автоматизованих функцій для проєктування етапів ливарної технології є надзвичайно актуальною.

До переваг даної розробки можна віднести:

- високу точність розрахунків і моделей;
- зменшення часу на проєктування;
- інтеграцію з вже існуючими системами SolidWorks;
- зручний та інтуїтивно зрозумілий інтерфейс користувача;
- можливість масштабування та адаптації під різні потреби виробництва.

Отже, актуальним стає завдання створення автоматизованих функцій для проєктування етапів ливарної технології за допомогою SolidWorks API.

Метою даної роботи є розробка елементів автоматизованого проєктування певних етапів ливарної технології за допомогою вбудованого в систему SolidWorks API з метою максимального спрощення та прискорення певних етапів проєктування.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- провести вивчення предметної галузі;
- провести огляд можливостей використання САПР SolidWorks для проєктування ливарної технології за допомогою 3D моделей;
- провести огляд методів автоматизації роботи в САПР SolidWorks за допомогою AddIn-ів;
- провести розробку модуля на базі даних з 3D моделями опок та інших компонентів ливарної технології;
- виконати проєктування програми;
- виконати реалізацію програми;
- виконати тестування.

Таким чином, у рамках дипломного проєкту розроблено автоматизовані функції, пов'язані з розрахунком та отриманням 3D моделей, що стосуються формувальних операцій процесу лиття. Практичне значення отриманих результатів полягає у створенні інженерного плагіну для програмного комплексу САПР SolidWorks, що надає можливість механізації та автоматизації проєктування технологічних процесів ливарного виробництва.

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ТА ПОСТАНОВКА ЗАВДАННЯ

1.1. Загальні відомості з предметної галузі

Лиття в піщано-глинисті форми є найбільш широко застосовуваним у машинобудуванні процесом отримання заготовок і становить від 60 до 90 % загального обсягу виробництва заготовок складної форми (рис. 1.1). Це зумовлено такими технологічними можливостями:

- форма відливок може бути досить складною, хоча й обмежена необхідністю вилучення моделі з форми;

- розміри відливки теоретично необмежені. Таким способом отримують найбільші відливки (до сотні тонн), приміром станини верстатів, корпуси турбін тощо;

- відносно низька вартість;

- висока продуктивність – до 180 – 240 форм за годину (на опокових автоматичних лініях) та до 500 форм за годину (безопокове формування);

- можливість механізації (ливарні конвеєри) та повної автоматизації процесу (автоматичні ливарні лінії) [1].



Рис. 1.1. Лиття в піщано-глинисті форми

Однак підготовчий етап, а саме проєктування ливарної технології традиційним способом є складним та трудомістким завданням. Стандартний цикл розробки технологічного процесу сталевих лиття в піщано-глинисті форми включає більше 20 розрахунково-аналітичних операцій [2], а саме:

- визначення площини роз'єму моделі та форми;
- визначення проливання отворів;
- розрахунок припусків на механічну обробку;
- визначення технологічних поповнень;
- розміщення ливарних радіусів і жолобників;
- визначення формувальних нахилів;
- розрахунок розмірів і розміщення прибутків;
- проєктування ливникової системи;
- визначення часу заливання форми й тривалості витримки відливків у формі;
- вибір типу формувальної суміші та фарби, визначення їх маси;
- вказівка режиму сушіння форм;
- визначення меж і конфігурації стрижнів із зазначенням типу й маси стрижневої суміші, ливарної фарби на кожен стрижень;
- вибір розмірів і конфігурації знаків стрижнів та зазорів між знаком стрижня й форми;
- вибір типу й розмірів каркасів стрижнів та газовідвідних елементів (при необхідності);
- вибір режиму сушіння стрижнів;
- визначення необхідності та місць встановлення жеребків;
- визначення потреби та місць встановлення зовнішніх і внутрішніх холодильників з розрахунком маси холодильника й маси «гарячого» вузла;
- проєктування розміщення моделей на підмодельній або координатній плиті;
- визначення конфігурації вставки або вимостки;
- визначення розмірів опок або ями;
- визначення заливальної ваги відливки;

– заповнення технологічного креслення та технологічних карт техпроцесу й виготовлення стрижнів.

Практично кожна з цих операцій пов'язана з необхідністю вибору даних із багатьох таблиць та графіків. Наприклад, тільки розрахунок припусків по ДСТУ 8981:2020 «Виливки з металів та сплавів. Допуски розмірів, маси та припуски на механічне оброблення» [3] пов'язаний з 15 таблицями, для кожного розміру необхідна робота з чотирма з них, що при ручному розрахунку для складної відливки займає у технолога невиправдано багато часу. Крім того, величезне різноманіття геометричних форм та розмірів відливок вимагає від технолога колосального досвіду й знань. Утім, у сучасних умовах спостерігається дефіцит якісних кадрів, що пов'язано з провалом у підготовці повноцінної зміни досвідчених кадрів.

З появою сучасних програмних пакетів інженерного 3D моделювання стало можливим використовувати 3D і для проектування ливарної технології. Зокрема, засобами САПР SolidWorks можливе створення повної 3D цифрової копії ливарної технології та всіх її елементів. Однак, незважаючи на безумовне підвищення наочності та зручності, стандартні засоби в САПР SolidWorks надають недостатнє прискорення процесу проектування. Крім того, з'являється низка додаткових вимог до технологів-проектувальників ливарних технологій – володіння загальними основами 3D моделювання та вміння користуватися САПР SolidWorks на рівні просунутого користувача.

Метою даної роботи є розробка елементів автоматизованого проектування певних етапів ливарної технології за допомогою вбудованого в систему SolidWorks API з метою максимального спрощення та прискорення певних етапів проектування. У рамках дипломного проєкту розроблено автоматизовані функції, пов'язані з розрахунком та отриманням 3D моделей, що стосуються формувальних операцій процесу лиття.

Формування – найбільш складна та трудомістка операція виробництва відливок у разових піщано-глинистих формах. Трудомісткість виготовлення ливарних форм складає 40 – 60 % від загальної трудомісткості отримання відливок.

В умовах масового та великосерійного виробництва дрібних і середніх за масою відливків застосовується машинне формування. Ручне формування знаходить застосування в індивідуальному та дрібносерійному виробництві, а також при виробництві великих відливків. Поняття «ручне формування» дещо застаріле, оскільки багато робіт (подача формувальної суміші, трамбування, вилучення моделей, поворот і переміщення опок) на даний час механізовані.

Розглянемо послідовність ручного формування для відливків деталі типу втулки [1]:

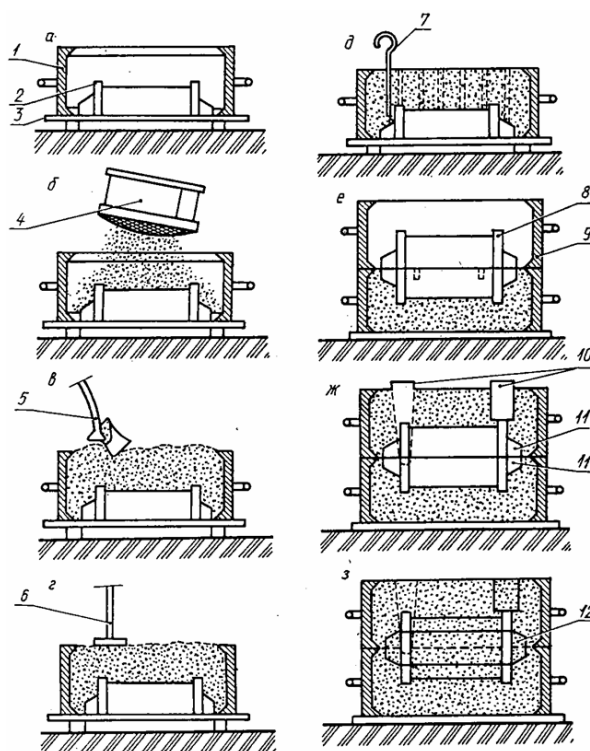


Рис. 1.2. Послідовність ручного формування для відливків деталі типу втулки

На підмодельний щит 3 (рис. 1.2, а) укладається половина моделі 2 і встановлюється нижня опока, потім крізь сито 4 на поверхню моделі наноситься протипригарний припил – деревно-вугільний пил, графітовий порошок (рис. 1.2, б). Лопатою 5 наносять на модель облицювальну формувальну суміш, а потім засипають всю опоку наповнювальною формувальною сумішшю (рис. 1.2, в). Ручним або пневматичним трамбуванням 6 ущільнюють суміш (рис. 1.2, г), згрібають її залишки і наколюють душником

(шилом) 7 отвори для кращого виходу газів (рис. 1.2, д). Потім нижню опоку із заформованою моделлю перевертають на 180°, встановлюють другу половину моделі 8 і верхню опоку 9 (рис. 1.2, е). Після установки моделей ливарної системи 10 у тій же послідовності заформовують верхню опоку (рис. 1.2, ж). Після закінчення формування опоки роз'єднують, обережно видаляють моделі, виправляють обвалені місця форми, припилюють її зсередини і, поклавши в нижню півформу на місце знаків 11 стрижень 12 (рис. 1.2, з), знову встановлюють верхню півформу на нижню і скріплюють їх за допомогою бовтів, струбцин або просто притискають вантажем, щоб запобігти прориву металу по площині роз'єму форми. У такому вигляді ливарна форма готова для заливання металу.

1.2. Призначення розробки та область застосування

Метою даної роботи є розробка інженерного плагіну для програмного комплексу САПР SolidWorks.

Результати кваліфікаційної роботи можуть бути використані у процесі проектування ливарної технології в системі SolidWorks для професійної діяльності.

Також результати роботи можна використовувати в освітньому процесі при вивченні відповідних дисциплін.

1.3. Підстави для розробки

Підставами для розробки (виконання кваліфікаційної роботи) є:

- освітня програма спеціальності 122 «Комп'ютерні науки»;
- навчальний план та графік навчального процесу;
- наказ ректора Національного технічного університету «Дніпровська політехніка» № 469-с від 23.05.2024 р.;
- завдання на кваліфікаційну роботу на тему: «Розробка інженерного плагіну для програмного комплексу САПР SolidWorks».

1.4. Постановка завдання

Метою даної роботи є розробка інженерного плагіну для програмного комплексу САПР SolidWorks.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- провести вивчення предметної галузі;
- провести огляд можливостей використання САПР SolidWorks для проєктування ливарної технології за допомогою 3D моделей;
- провести огляд методів автоматизації роботи в САПР SolidWorks за допомогою AddIn-ів;
- провести розробку модуля на базі даних з 3D моделями опок та інших компонентів ливарної технології;
- виконати проєктування програми (плагіну);
- виконати реалізацію програми (плагіну);
- виконати тестування плагіну.

1.5. Вимоги до програми або програмного виробу

1.5.1. Вимоги до функціональних характеристик

Під час проєктування було визначено функціональні та нефункціональні вимоги до системи, що розробляється.

Функціональні вимоги визначають функціональність програмного забезпечення, тобто описують, яку поведінку повинна надавати система, що розробляється.

Система, що розробляється, повинна задовольняти наступним функціональним вимогам:

- система повинна надавати можливість завантаження файлів 3D моделей технології формату SolidWorks 2024 або попередніх версій розміром до 100 МБ;
- система повинна автоматично зберігати результати роботи в файлах формату SolidWorks 2024;
- система повинна надавати можливість редагування параметрів опок у базі даних;

- база даних системи має надавати можливість зберігання 3D моделей опок розміром до 50 МБ;

- база даних системи має надавати можливість зберігання до 200 моделей опок та їх параметрів;

- система має надавати можливість вказувати комплекти 3D моделей опок для установки;

- система має надавати можливість вказувати місця для установки випорів на 3D моделі технології;

До нефункціональних вимог системи відносяться властивості, які вона повинна мати. Наприклад, зручність використання, безпека, розширюваність тощо.

Система повинна задовольняти такі нефункціональні вимоги:

- система має бути написана мовою C++ [10];

- система повинна використовувати API SolidWorks 2024 [6] для роботи AddIn модулю;

- модуль бази даних та діалоги користувача повинні бути розроблені у вигляді окремої DLL.

1.5.2. Вимоги до інформаційної безпеки

Для уникнення некоректної роботи програми необхідно реалізувати:

- контроль вхідних даних;

- обробку виняткових ситуацій;

- можливість повторного введення даних;

- виведення повідомлень про помилки;

- можливість безперервної роботи протягом не менше 8 годин;

- забезпечення збереження та неушкодженого стану даних, що зберігаються в базі даних, у випадку відмови програми.

Особливих чи додаткових вимог до інформаційної безпеки додатку не висувається.

1.5.3. Вимоги до складу та параметрів технічних засобів

Для нормального зручного користування та функціонування програми необхідно, щоб обчислювальна машина, на якій буде функціонувати програма, відповідала наступним вимогам:

- центральний процесор архітектури x86-64 з підтримкою розширення команд SSE2, з тактовою частотою не менш 1.0 ГГц;
- графічний прискорювач з підтримкою API OpenGL 4.4 та оперативною пам'яттю 1 GB;
- не менше 8 GB оперативної пам'яті;
- монітор з роздільною здатністю 1080p (Full HD);
- 2 Гб вільного місця на жорсткому диску;
- клавіатура;
- маніпулятор «миша».

Наведені вище технічні характеристики є рекомендованими, тобто при наявності технічних засобів не нижче зазначених, розроблений програмний виріб буде функціонувати відповідно до вимог щодо надійності, швидкості обробки даних і безпеки, висунутими замовником.

1.5.4. Вимоги до інформаційної та програмної сумісності

Для нормального функціонування програми у користувача необхідно, щоб на обчислювальній машині було встановлено наступне програмне забезпечення:

- операційна система Windows 10 x64;
- система автоматизованого параметричного 3D проектування SolidWorks 2024 версії Standard або вище;
- система керування базами даних Firebird 5.0 x64.

РОЗДІЛ 2

ПРОЄКТУВАННЯ ТА РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ

2.1. Функціональне призначення системи

Система, що розробляється, повинна задовольняти наступним функціональним вимогам:

- здійснювати розрахунок розмірів та вставку опок з бази попередньо сформованих 3D моделей для будь-якої попередньо сформованої 3D моделі ливарної технології в діапазоні розмірів, відповідному введеним у базу розмірам опок;

- у разі відсутності в базі даних комплекту 3D моделей верхніх та нижніх опок підхожого розміру виводити відповідне повідомлення й переривати операцію;

- забезпечувати користувачеві можливість вибору комплекту 3D моделей опок зі списку відповідних за розміром в окремому діалоговому вікні;

- розраховувати і створювати 3D моделі верхньої та нижньої півформ, що відповідають розмірам обраних опок й ливарної технології та є еквівалентами фізичних півформ із формувальної суміші;

- забезпечувати зберігання та вибірку попередньо створених стандартних 3D моделей опок;

- забезпечувати редагування списку моделей опок, включаючи геометричні параметри та 3D модель;

- створювати 3D моделі верхньої та нижньої формувальних півмоделей, придатних для виготовлення вручну або на верстатах з ЧПУ;

- розраховувати розміри, генерувати та вставляти 3D моделі випорів у вказану користувачем точку на 3D моделі;

- розраховувати та виводити в окреме інформаційне вікно основні результуючі технологічні параметри – загальну масу ливарної збірки, масу формувальної суміші, масу та об'єм покриття.

2.2. Опис застосованих математичних методів

Згідно вимог до системи, що розробляється, одним з результатів роботи є розрахунки загальної маси ливарної збірки, маси формувальної суміші, маси та об'єму покриття. В концепції системи ці розрахунки мають проводитись на основі відповідних згенерованих 3D моделей, тож для цього потрібно застосувати методи розрахунків площі поверхні та об'єму тіл 3D моделей, що представлено у B-реп (Boundary representation) вигляді.

Розрахунок площі поверхні тіла. Площа поверхні тіла складається з площ поверхні його граней. Площа S кожної поверхні визначається інтегралом:

$$S = \iint_S dS = \iint_{\Omega} \sqrt{g_{11}g_{22} - g_{12}^2} dudv. \quad (2.1)$$

Площа поверхні тіла дорівнює сумі площ граней:

$$S = \sum_i \iint_{S_i} dS = \sum_i \iint_{\Omega_i} \sqrt{g_{11}g_{22} - g_{12}^2} du_i dv_i, \quad (2.2)$$

де u_i, v_i – параметри поверхні i -ої грані тіла;

Ω_i – область визначення параметрів поверхні i -ої грані тіла.

У подальших формулах геометричних характеристик тіла ми опустимо знак додавання та індекси граней, вважаючи, що додавання, звісно, виконується.

Розрахунок об'єму тіла. Якщо об'єм тіла V є обмеженим і просторово однозв'язним, поверхня тіла S – замкнутою і регулярною, а функції $F(r)$ та $\varphi(r)$ – однозначними, безперервними й такими, що мають безперервні приватні похідні всередині оболонки тіла і на її поверхні, то поверхневі інтеграли

пов'язані з інтегралами по об'єму тіла відповідно теоремою Остроградського-Гаусса (теоремою про дивергенцію):

$$\iint_S m \cdot F(r) dS = \iiint_V \nabla \cdot F(r) dV. \quad (2.3)$$

Теорема Остроградського-Гаусса свідчить, що інтеграл від дивергенції векторного поля по деякому об'єму дорівнює потоку поля через поверхню, яка обмежує цей об'єм. Векторна функція $F(r)$ в (2.3) сама може бути отримана за участю оператора, наприклад, $F(r) = \nabla \varphi(r)$, $F(r) = \psi \nabla \varphi(r)$, або $F(r) = \psi \nabla \varphi(r) - \varphi \nabla \psi(r)$.

Припустимо в (2.3) $F(r)=r$. Тоді $\nabla \cdot (r)=3$ і з теореми Остроградського-Гаусса отримаємо:

$$\iint_S m \cdot r dS = 3 \iiint_V dV = 3V.$$

Таким чином, об'єм тіла визначається за формулою:

$$V = \frac{1}{3} \iint_S m \cdot r dS = \frac{1}{3} \iiint_{\Omega} m \cdot r \sqrt{g_{11}g_{22} - g_{12}^2} dudv. \quad (2.4)$$

Нехай щільність тіла ρ постійна по всьому його об'єму. Тоді маса тіла визначиться за формулою:

$$M = \rho V = \frac{\rho}{3} \iint_S m \cdot r dS = \frac{\rho}{3} \iiint_{\Omega} m \cdot r \sqrt{g_{11}g_{22} - g_{12}^2} dudv. \quad (2.5)$$

Якщо щільність тіла не є постійною, то нам не вдасться перейти від інтегралу по об'єму до інтегралу по поверхні тіла при обчисленні його маси. У

формулах (2.4) та (2.5) мається на увазі додавання по граням тіла. Зауважимо, що m є нормаль грані тіла, а не поверхні цієї грані. Нормаль грані завжди направлена назовні тіла, а нормаль поверхні, на якій базується грань, може збігатися з нормаллю грані або може мати протилежний нормалі грані, напрямком. Загалом тіло може мати порожнечі всередині. Тоді воно описується кількома замкнутими оболонками, одна з яких є зовнішньою, а решта внутрішніми і повністю лежать усередині зовнішньої оболонки. Нормаль до будь-якої оболонки направлена поза об'ємом тіла. Інтегрування (2.4) і (2.5) виконується по всіх оболонках, як зовнішньої оболонки, так і внутрішнім оболонкам.

Формулу (2.4) можна отримати й іншим способом. Розглянемо нескінченно малу ділянку dS поверхні деякої грані тіла. Побудуємо елементарну піраміду на базі нескінченно малої ділянки dS з вершиною на початку координат і прямолінійними утворюючими. Таку елементарну піраміду наведено на рис. 2.1. Основою піраміди є розглянута нескінченно мала ділянка dS поверхні.

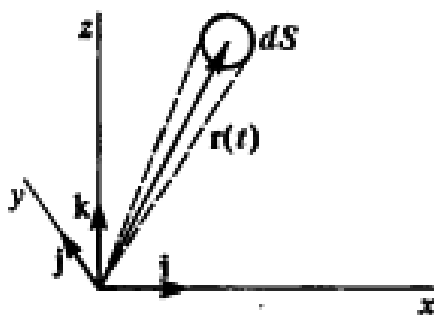


Рис. 2.1. Піраміда з нескінченно малою основою dS

Нехай поверхня розглянутої грані описується векторною функцією $r = r(u, v)$. Розглянемо вираз:

$$\frac{1}{3} m \cdot r dS, \quad (2.6)$$

де m – одиничний вектор нормалі розглянутої поверхні.

Абсолютна величина виразу $m \cdot r dS$ дорівнює об'єму елементарної піраміди, побудованої на ділянці поверхні dS , так як модуль $m \cdot r$ представляє собою висоту, опущену з вершини піраміди на її основу. Знак $m \cdot r dS$ залежить від орієнтації нормалі поверхні та її радіус-вектора. Якщо обчислити інтеграл від виразу (2.6) по всіх гранях тіла, то в силу того, що оболонка тіла завжди є замкнутою і орієнтованою, отримаємо об'єм цього тіла.

2.3. Опис використаних технологій та мов програмування

Для розробки системи використано високорівневу мову програмування C++, САПР SolidWorks та SolidWork API.

C++ - статично типізована мова програмування загального призначення, з компіляцією в нативний код цільової платформи.

Підтримує такі парадигми програмування, як процедурне програмування, об'єктно-орієнтоване програмування, узагальнене програмування. Мова має багату стандартну бібліотеку, яка включає поширені контейнери і алгоритми, введення-виведення, регулярні висловлювання, підтримку багатопоточності та інші можливості. C++ поєднує властивості як високорівневих, і низькорівневих мов.

САПР SolidWork та її API. Програмний продукт SolidWorks є найпоширенішим інструментом для автоматизованого проектування (САПР) і 3D моделювання. Пакет дозволяє створити деталі для майбутнього 3D друку.

Особливості SolidWorks [4]:

- спеціальна бібліотека з різноманітними стандартними компонентами та виробами, що містить деталі з різним призначенням та параметрами. В ній можна швидко знайти будь-який об'єкт або скористатися стандартним компонентом як деталлю для модифікації;

- прив'язки. Це особливість програми, що дозволяє виконувати прив'язки прямо в моделі. При цьому робочі елементи предмета, що проектується,

зв'язуються між собою для запобігання спотворенню побудованого об'єкта при внесенні необхідних змін до самої моделі;

- робота з різними рівняннями. У програмі є можливість зв'язування всієї моделі за допомогою рівнянь, винесених в окремий створений документ текстового формату. При цьому при зміні одного елемента в об'єкті відбудеться синхронна зміна усієї моделі;

- автоматична побудова креслення прямо з моделі. Така особливість практично незамінна для проєктування різноманітних технічних виробів. Отримати креслення з моделі можна лише за кілька кліків комп'ютерною мишею;

- наявність можливості зшивання поверхонь. Це дозволяє отримати твердотільну модель, яка стане придатною для відтворення на 3D принтері.

Можливості програми SolidWorks. Базові функції цієї програми доступні користувачам без застосування спеціальних розширень, але створені додаткові модулі збільшують її функції.

Деякі можливості цієї програми:

- 3D моделювання різних твердих тіл;
- розробка необхідних зварних конструкцій;
- проведення розрахунків на міцність, точний розрахунок гідро- та аеродинаміки;
- створення різноманітних креслень;
- виконання проєктування з урахуванням матеріалу, що використовується для створення виробу;
- візуалізація та проведення розрахунку на можливий вигин;
- використання даних після виконаного 3D сканування;
- проєктування різноманітної продукції для виготовлення якої використовується листовий метал;
- читання електросхем;
- експорт отриманих даних та їх переведення у різні формати.

Переваги та недоліки програми SolidWorks. Програма SolidWorks для автоматизованого проєктування має такі переваги:

- підготовка виробництва різноманітних виробів незалежно від їх складності та функціонального призначення;
- відсутність обмежень щодо кількості компонентів складних збірок;
- широкі можливості для оформлення необхідної конструкторської документації;
- можливість випробування спроектованих моделей на міцність у наближених до реальних умов;
- відмінне співвідношення ціни і продуктивності;
- просте використання та підтримка кількох стандартів.

Недоліком цього програмного забезпечення є те, що інколи зустрічається некоректне автоматичне проставлення розмірів, що легко виправляється користувачем вручну[5].

API SolidWorks та його можливості. Інтерфейс прикладного програмування SolidWorks (англ. Application Programming Interface, API) – це COM-інтерфейс програмування для програмного забезпечення SolidWorks. Функції API надають програмістам прямий доступ до функцій SolidWorks.

API містить сотні функцій, які можна викликати з файлів макросів Visual Basic (VB), Visual Basic for Applications (VBA), VB.NET, C++, C# або SolidWorks.

Об'єкт ISldWorks є об'єктом додатка; це об'єкт верхнього рівня в ієрархії об'єктів SolidWorks API. Усі інші об'єкти SolidWorks розташовані нижче об'єкта ISldWorks в ієрархії об'єктів, і до них можна отримати прямий чи опосередкований доступ (рис. 2.2) [6].

Якщо доступ до об'єкта можливий лише опосередковано, тоді потрібно посилатися на цей об'єкт з іншого об'єкта, розташованого вище в ієрархії об'єктів. Наприклад, доступ до об'єкта ISketchSpline можливий лише опосередковано; він не може існувати сам по собі. Повинне посилання на

об'єкт ISketchSpline із об'єкта ISketchSegment, оскільки він існує тільки в контексті об'єкта ISketchSegment.

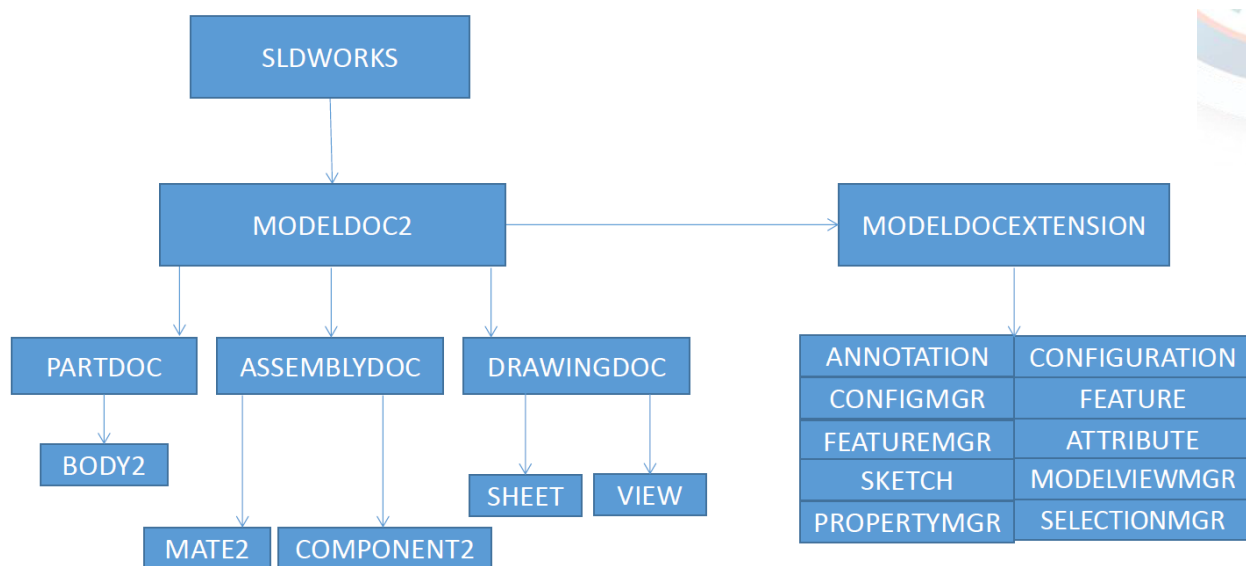


Рис. 2.2. Структурна схема основних об'єктів API SolidWorks

Більшість об'єктів SolidWorks API відповідають функціям інтерфейсу користувача; однак деякі об'єкти SolidWorks API надають функціональні можливості, доступні тільки через SolidWorks API. Наприклад, об'єкт IFeature дозволяє отримати доступ до елементів у дереві конструювання FeatureManager, але об'єкт IAttribute є об'єктом тільки SolidWorks API, оскільки не має відповідних функцій інтерфейсу користувача.

Використання API для автоматизації SolidWorks виконується через розробку Add-in додатків.

Add-in SolidWorks – це додатки для SolidWorks, які забезпечують високий рівень налаштування, а також максимальну продуктивність і зручність використання SolidWorks API. Add-in можна завантажувати під час запуску та інтегрувати в додаток, надаючи можливість розширення інтерфейсу користувача і функціональності SolidWorks.

Add-in SolidWorks – це COM-об'єкт, який може бути розроблений з використанням будь-яких COM-сумісних мов, таких як C++, C#, VB.NET, VB6.

Для розробки add-in необхідно:

- створити публічний COM-видимий клас, який реалізує інтерфейс ISwAddIn;
- зареєструвати dll як COM-сервер за допомогою Regasm для add-in на основі .NET (C# або VB.NET) і RegSvr32 для add-in на основі C++ і VB6. Під час реєстрації COM-об'єкта необхідно додати кодову базу dll (рис. 2.3).

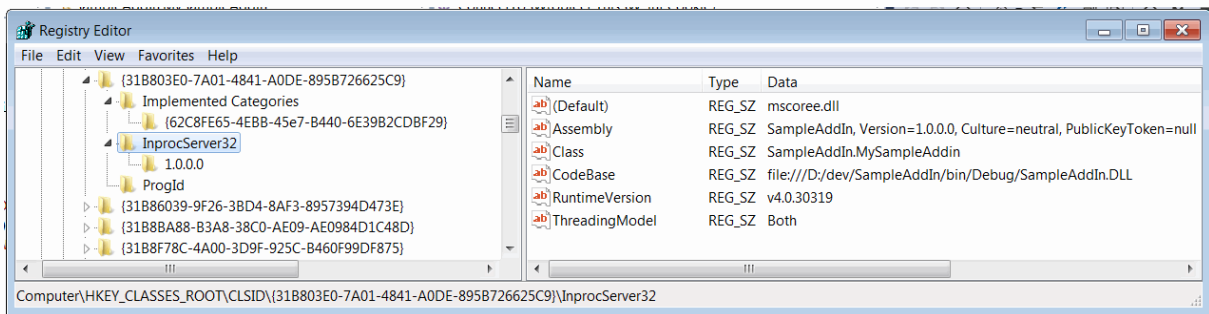


Рис. 2.3. Параметри реєстру необхідні для реєстрації Add-in SolidWorks

Інтерфейс ISwAddIn надає дві функції для реалізації:

1. **ConnectToSW**. Викликається, коли add-in завантажується за допомогою SolidWorks (якщо вказано параметр завантаження під час запуску) або коли вона включена в меню Інструменти->Add-in в SolidWorks (Tools->AddIns menu in SolidWorks). Функція передає 2 параметри: вказівник на основний інтерфейс ISldWorks та ідентифікатор add-in.

Вказівник на ISldWorks використовується в коді для доступу до всіх інтерфейсів. Другий параметр – це ідентифікатор add-in, який можна використовувати для додавання компонентів інтерфейсу користувача в такі методи, як ISldWorks::GetCommandManager, ISldWorks::AddFileSaveAsItem2, ISldWorks::ShowBubbleTooltipAt2. Використовується для ініціалізації ресурсів та створення інтерфейсу користувача (диспетчера команд, панелі завдань тощо).

2. **DisconnectFromSW**. Викликається, коли add-in вивантажується під час закриття SolidWorks або при вимкненні в меню Інструменти->Add-in в SolidWorks (Tools->AddIns menu in SolidWorks). Використовується цей метод, щоб очистити ресурс і видалити всі елементи інтерфейсу користувача (наприклад, диспетчер команд або панель завдань).

Він імпортується для вивантаження сторінок інтерфейсу користувача, оскільки користувач може вивантажити add-in, відключивши її в меню add-in (не лише закривши додаток SolidWorks), і збереження завантажених сторінок інтерфейсу користувача може створити ризик збою або нестабільності, оскільки сторінка відключається від самої add-in.

FIREBIRD DATABASE. Firebird Database – потужна і гнучка система управління базами даних (СУБД), що має відкритий вихідний код й широко використовується в різних галузях[8]. Нижче представлена інформація про її особливості, архітектуру, можливості та переваги.

1. Основні характеристики.

Відкритий вихідний код: Firebird Database розповсюджується під ліцензією Initial Developer's Public License (IDPL), що дозволяє користувачам вільно поширювати, модифікувати та використовувати вихідний код без необхідності сплачувати ліцензійні збори.

Мультиплатформність: Firebird підтримує різні операційні системи, включаючи Windows, Linux та macOS, що робить його зручним вибором для розробників на різних платформах.

Транзакційна безпека: Firebird забезпечує високий рівень транзакційної безпеки, підтримуючи ACID (атомарність, узгодженість, ізольованість, стійкість) узгодженість даних.

Масштабованість: Firebird підтримує як невеликі локальні бази даних, так і великі розподілені системи з сотнями користувачів та серверів.

2. Архітектура та компоненти.

Серверна частина: Firebird працює у вигляді сервера баз даних, що управляє доступом до даних та виконанням запитів. Серверна частина може працювати в різних режимах доступу до даних, включаючи класичний, суперсервер та суперклієнт.

Клієнтські бібліотеки: Firebird надає клієнтські бібліотеки для взаємодії із сервером баз даних. Ці бібліотеки є доступними для різних мов програмування, таких як Java, C++, Python та інших, що забезпечує широку підтримку для розробників.

Утиліти адміністрування: Firebird включає різні утиліти для управління базою даних, включаючи утиліти резервного копіювання, відновлення, моніторингу і т.д.

3. Можливості та переваги.

Підтримка зберігання процедур і тригерів: Firebird підтримує створення процедур і тригерів, які зберігаються на стороні сервера, що дозволяє розробникам реалізовувати складну бізнес-логіку в базі даних.

Підтримка стандартів SQL: Firebird відповідає стандартам SQL, що забезпечує сумісність з існуючими додатками та інструментами, які підтримують SQL.

Висока продуктивність: Firebird забезпечує високу продуктивність при обробці транзакцій та виконанні запитів до бази даних, що робить його підходящим вибором для багатьох додатків із високими вимогами до продуктивності.

Ефективне використання ресурсів: Firebird ефективно використовує системні ресурси, що дозволяє заощаджувати місце на диску, використовувати менше оперативної пам'яті та забезпечувати більш надійну роботу бази даних.

4. Особливості архітектури.

Багаторівнева архітектура: Firebird використовує багаторівневу архітектуру, яка включає різні шари, такі як рівень додатку, серверний рівень та рівень сховища даних. Це дозволяє легко масштабувати й обслуговувати бази даних різного розміру та складності.

Асинхронний запис транзакцій: Firebird використовує асинхронний запис транзакцій, що підвищує продуктивність та відмовостійкість бази даних. Це означає, що дані записуються на диск асинхронно від виконання запитів, що зменшує навантаження на сервер під час роботи з базою даних.

Механізми реплікації та кластеризації: Firebird підтримує механізми реплікації та кластеризації, які дозволяють створювати розподілені системи з кількома репліками бази даних для підвищення відмовостійкості й масштабованості[9].

5. Інструменти розробки та адміністрування.

Firebird Development Studio: це інтегроване середовище розробки (IDE), що надає широкий спектр інструментів для розроблення баз даних Firebird, включаючи редактор SQL, візуальний конструктор запитів, інструменти для управління об'єктами бази даних і багато іншого.

IBExpert: це ще одна популярна IDE для роботи з Firebird, що надає розширені інструменти для адміністрування, розробки та оптимізації баз даних.

6. Переваги Firebird Database.

Безкоштовність та відкритий вихідний код: Firebird розповсюджується безкоштовно і має відкритий вихідний код, що робить його доступним для широкого кола користувачів та забезпечує високу ступінь гнучкості й налаштовуваності.

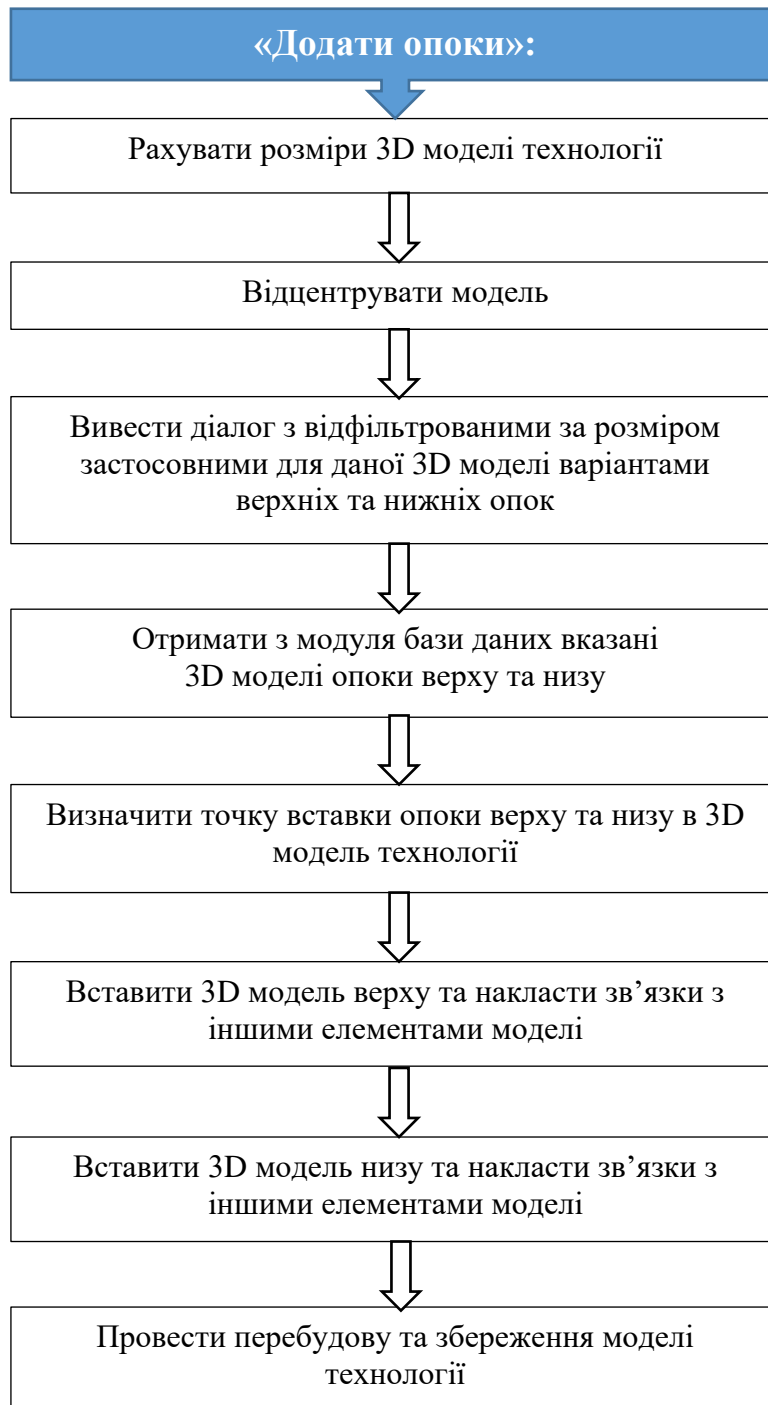
Активна спільнота: Firebird має активну спільноту розробників та користувачів, яка забезпечує підтримку, оновлення та нові функції для продукту.

Надійність та стабільність: Firebird є надійною та стабільною СУБД, перевіреною роками використання у різних критичних додатках і системах.

2.4. Опис структури системи та алгоритмів її функціонування

Система складається з двох модулів – модулю плагіна SolidWorks та модулю бази даних. Так, плагін SolidWorks реалізує роботу команд, пов'язаних зі зчитуванням і зміною 3D моделей, використовуючи API SolidWorks. Модуль бази даних забезпечує зберігання та вибірку даних, необхідних для побудови елементів 3D моделей (розміри, константи для розрахунків, матеріали тощо), а також зберігання та видачу 3D моделей стандартних елементів технології (опоки, підмодельні плити).

Алгоритми команд (функцій) модуля SolidWorks включають (рис. 2.4):



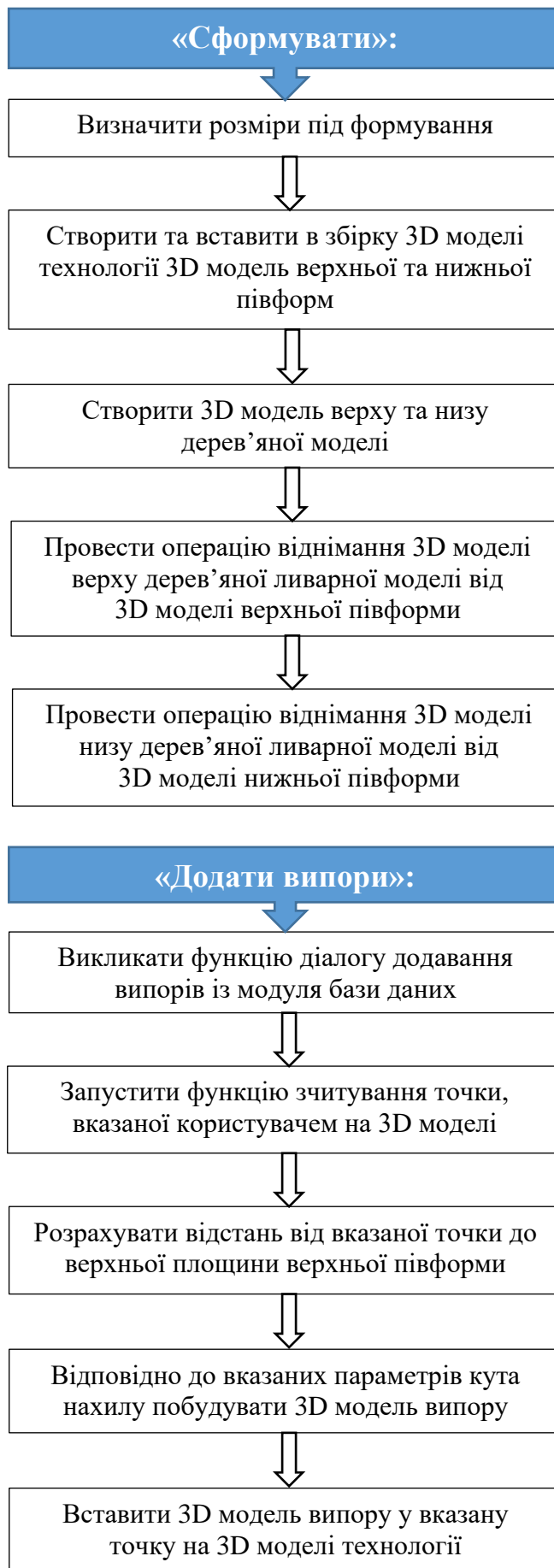


Рис. 2.4. Алгоритми команд «Додати опоки», «Сформувати», «Додати випори» модуля SolidWorks

2.5. Обґрунтування та організація вхідних та вихідних даних програми

Проектування формування є одним із заключних етапів у розробці технології лиття, відповідно для його виконання має бути попередньо сформована 3D модель, що містить результати проектування попередніх етапів. Для операції підбору опок при проектуванні технології необхідно:

- знати та враховувати їх геометричні параметри (довжина, ширина, висота) і мати власне 3D моделі відповідних розмірів та форм;
- враховувати мінімально допустимі відстані від відливки до опоки по всіх трьох вимірах. Ці відстані залежать від виду формування (машинна чи ручна), задаються методикою розрахунку і залежать від діапазонів розмірів відливок.

Крім того, у процесі проектування виконується низка розрахунків, для яких використовуються певні константи в залежності від розмірів, форми та інших параметрів лиття.

Таким чином, для роботи системи необхідні такі вхідні дані:

1. Попередньо сформована 3D модель ливарної технології, що включає 3D моделі наступних компонентів:

- відливка;
- первинна деталь;
- формувальна модель;
- прибутки;
- ливарна система;
- стрижні (опційно).

2. База 3D моделей опок з їх геометричними параметрами.

3. Таблиця відстаней від відливки до опоки по всіх трьох вимірах для машинного формування.

4. Таблиця відстаней від відливки до опоки по всіх трьох вимірах для ручного формування.

У 3D моделі технології має бути вказана площина роз'єму форми (рис. 2.5).

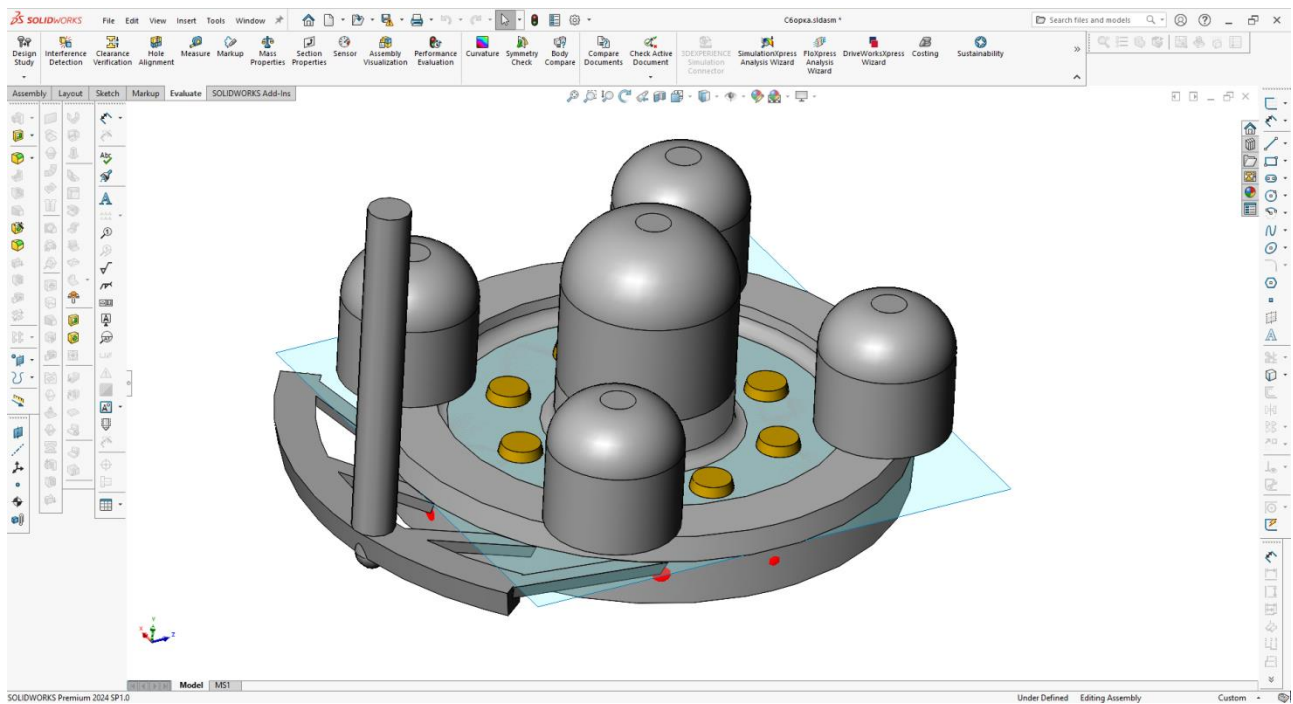


Рис. 2.5. Вихідна 3D модель технології з площиною роз'єму форми

Для зберігання, редагування та обробки даної інформації було створено базу даних Firebird з відповідними таблицями. У таблиці опок вноситься та зберігається інформація про геометричні характеристики опок у вигляді цифрових полів та 3D моделі опок у вигляді BLOB об'єктів (рис. 2.6, 2.7).

#	PK	FK	UNQ	Field Name	Field Type	Domain	Size	Scale	Subtype	Array	Not Null	Charset	Collate	Description
1				NPP	INTEGER						<input type="checkbox"/>			
2				NAIMENOVANIE	CHAR		48				<input type="checkbox"/>	WIN1251	WIN1251	
3				TIP1	INTEGER						<input type="checkbox"/>			
4				TIP2	INTEGER						<input type="checkbox"/>			
5				DLINA	INTEGER						<input type="checkbox"/>			
6				SHIRINA	INTEGER						<input type="checkbox"/>			
7				VISOTA	INTEGER						<input type="checkbox"/>			
8				VES	NUMERIC		15	3			<input type="checkbox"/>			
9				KOL	INTEGER						<input type="checkbox"/>			
10				DETAIL	BLOB		80		Binary		<input type="checkbox"/>			

Рис. 2.6. Структура таблиці з параметрами та 3D моделями опок

Nr	Name	Length	Width	Height	Weight	Quantity	3D Model
33	Up flask	1200	1200	200		0	1 (BLOB)
34	Down flask	1200	1200	200		0	1 (BLOB)
35	Up flask	1200	1200	500		0	1 (BLOB)
36	Down flask	1200	1200	500		0	1 (BLOB)
37	Up flask	3000	2000	800		0	1 (BLOB)
38	Down flask	3000	2000	800		0	1 (BLOB)
39	Up flask	400	300	75		0	1 (BLOB)
40	Down flask	400	300	75		0	1 (BLOB)
41	Up flask	400	300	150		0	1 (BLOB)
42	Down flask	400	300	150		0	1 (BLOB)
43	Up flask	400	300	200		0	1 (BLOB)
44	Down flask	400	300	200		0	1 (BLOB)
50	Up flask	1200	1200	250		0	1 (BLOB)
46	Down flask	500	400	75		0	1 (BLOB)
47	Up flask	500	400	75		0	1 (BLOB)
48	Up flask	500	400	150		0	1 (BLOB)
49	Down flask	500	400	150		0	1 (BLOB)
51	Down flask	1200	1200	250		0	1 (BLOB)
52	Up flask	500	400	200		0	1 (BLOB)
53	Down flask	500	400	200		0	1 (BLOB)
54	Up flask	600	500	150		0	1 (BLOB)
55	Down flask	600	500	150		0	1 (BLOB)
56	Up flask	500	400	200		0	1 (BLOB)
57	Down flask	500	400	200		0	1 (BLOB)
58	Up flask	600	500	300		0	1 (BLOB)
59	Down flask	600	500	300		0	1 (BLOB)
60	Up flask	700	600	200		0	1 (BLOB)
61	Down flask	700	600	200		0	1 (BLOB)
62	Up flask	700	600	400		0	1 (BLOB)
63	Down flask	700	600	400		0	1 (BLOB)

Рис. 2.7. Заповнена таблиця з параметрами та 3D моделями опок

Також для заповнення бази даних опок було розроблено 3D моделі опок різної форми та типорозмірів – круглих (рис. 2.8), квадратних (рис. 2.9) та прямокутних (рис. 2.10, 2.11).

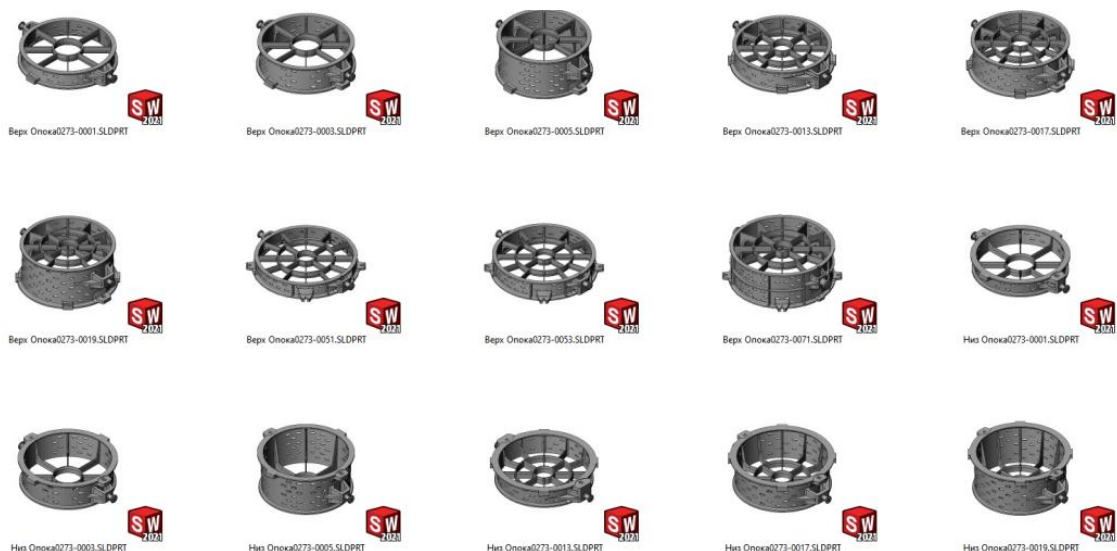


Рис. 2.8. Набір 3D моделей круглих опок, внесених у базу даних

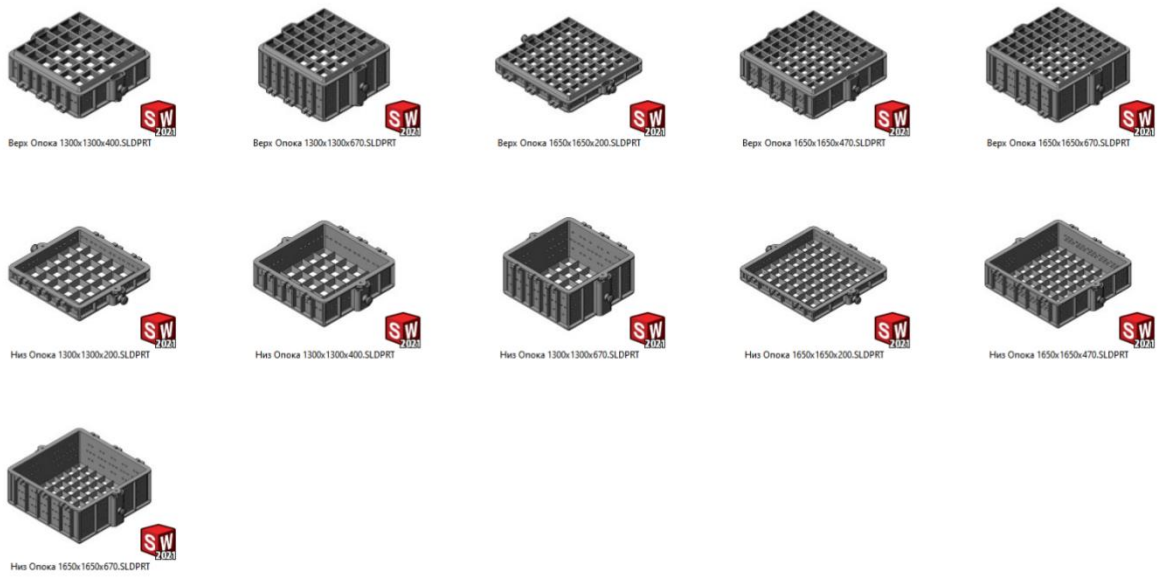


Рис. 2.9. Набір 3D моделей квадратних опок, внесених у базу даних

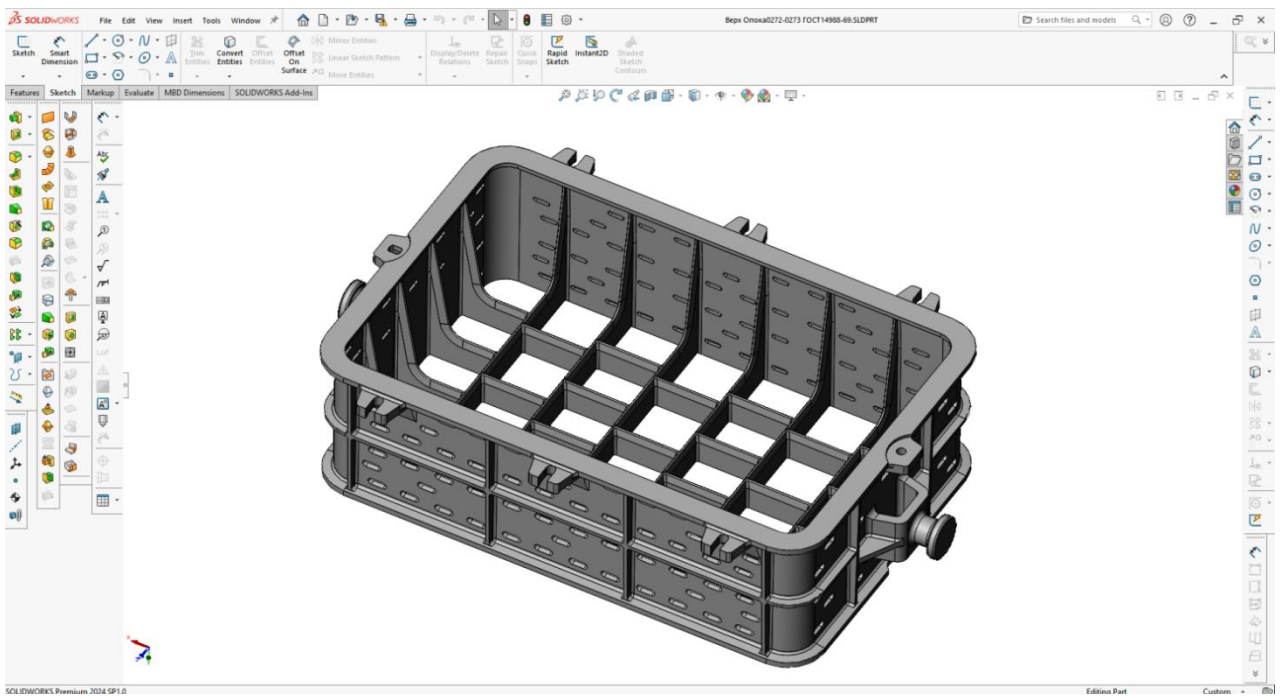


Рис. 2.10. Деталізована 3D модель прямокутної опоки



Рис. 2.11. Набір 3D моделей прямокутних опок, внесених у базу даних

Вихідними даними програми є модифікована 3D модель ливарної технології з доданими 3D моделями опок, півформ, формувальних моделей і випорів, а також результат розрахунку відповідних технологічних параметрів – загальної маси ливарної збірки, маси суміші, маси та об’єму покриття.

2.6. Опис роботи розробленої системи

2.6.1. Використані технічні засоби

Для написання кваліфікаційної роботи та прогорамного забезпечення використовувався ноутбук Ноутбук Lenovo Legion з наступними характеристиками: процесор AMD Ryzen 7 7 5800 H with Radeon Graphics CPU @4.45GHz, ОЗУ 32 ГБ, жорсткий диск SSD, графічний прискорювач NVIDIA GeForce RTX 3070 Laptop GPU.

2.6.2. Використані програмні засоби

Для розробки програми було вибрано високорівневу мову програмування C++ стандарту C++17.

Розробка Add-in для SolidWorks велася в IDE Microsoft Visual Studio Community 2022 (64-bit) 17.9.6

Для розробки DLL модулю бази даних та діалогів користувача було використано IDE Embarcadero C++ Builder 11.3 Community Edition.

Для розробки бази даних було використано СУБД Firebird 5.0 та GUI редактор баз даних IVExpert 2024.04.07.

2.6.3. Виклик та завантаження програми

Для виклику програми необхідно запустити систему SolidWorks 2024 та впевнитись, що Add-in модуль «Casting tech» активний. Для цього треба викликати діалог Add-Ins в меню Tools системи SolidWorks та поставити галочку в колонці «Active» на відповідному «Casting tech» рядку. При потребі в автоматичній загрузці Add-in модулю «Casting tech» при запуску системи SolidWorks потрібно також поставити галочку в колонці «Start Up». Програма може завантажуватись на ПК та ноутбуках, на яких встановлено Windows 10 x64, Firebird 5.0 x64 та SolidWorks 2024 та які відповідають параметрам технічних засобів, що наведено в п 1.5.3.

2.6.4. Опис інтерфейсу користувача

Основна взаємодія з користувачем відбувається через команди меню, інтегрованого в систему SolidWorks і діалоги, що викликаються через них. Розрахунок розмірів опок та вставки їх у 3D модель технології викликається через команду меню «Додати опоки» (рис. 2.12).

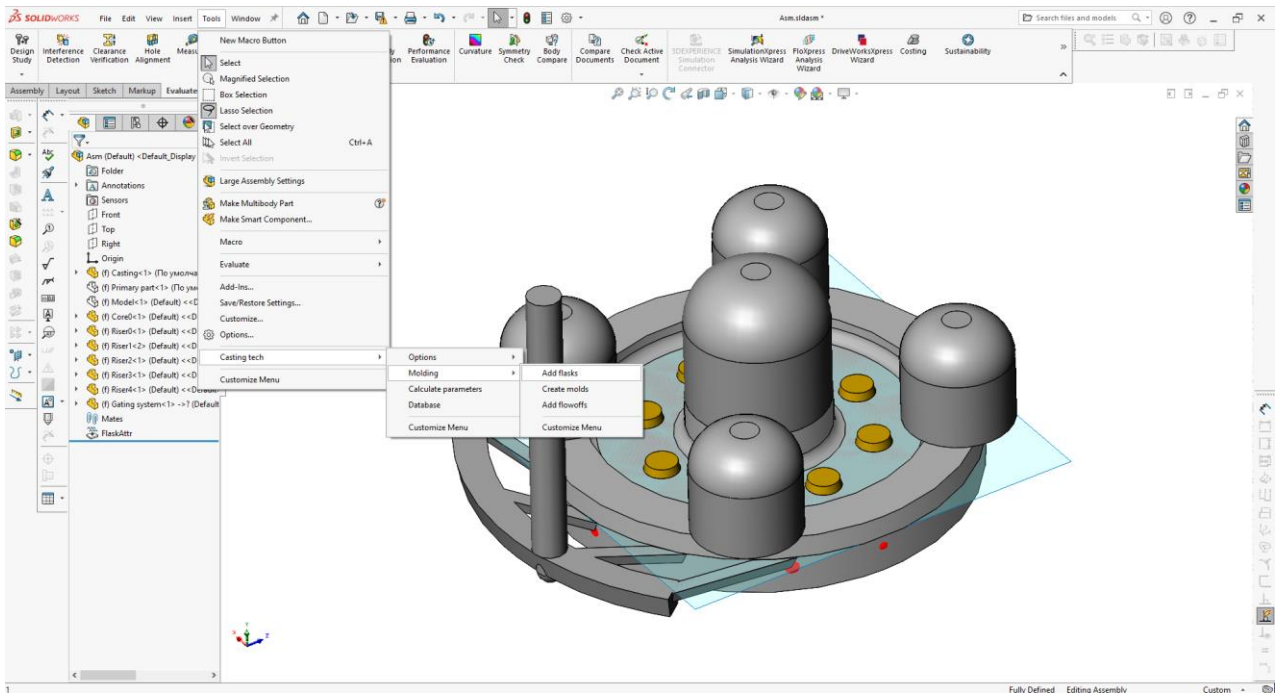


Рис. 2.12. Команда меню «Додати опоки»

Після виклику цієї команди користувачу відображається діалог вибору опок зі списку підходящих за розмірами. У цьому діалозі необхідно вказати опоку низу, після чого автоматично оновлюється список сумісних опок верху, з яких також необхідно вибрати одну і натиснути кнопку «Додати» (рис. 2.13).

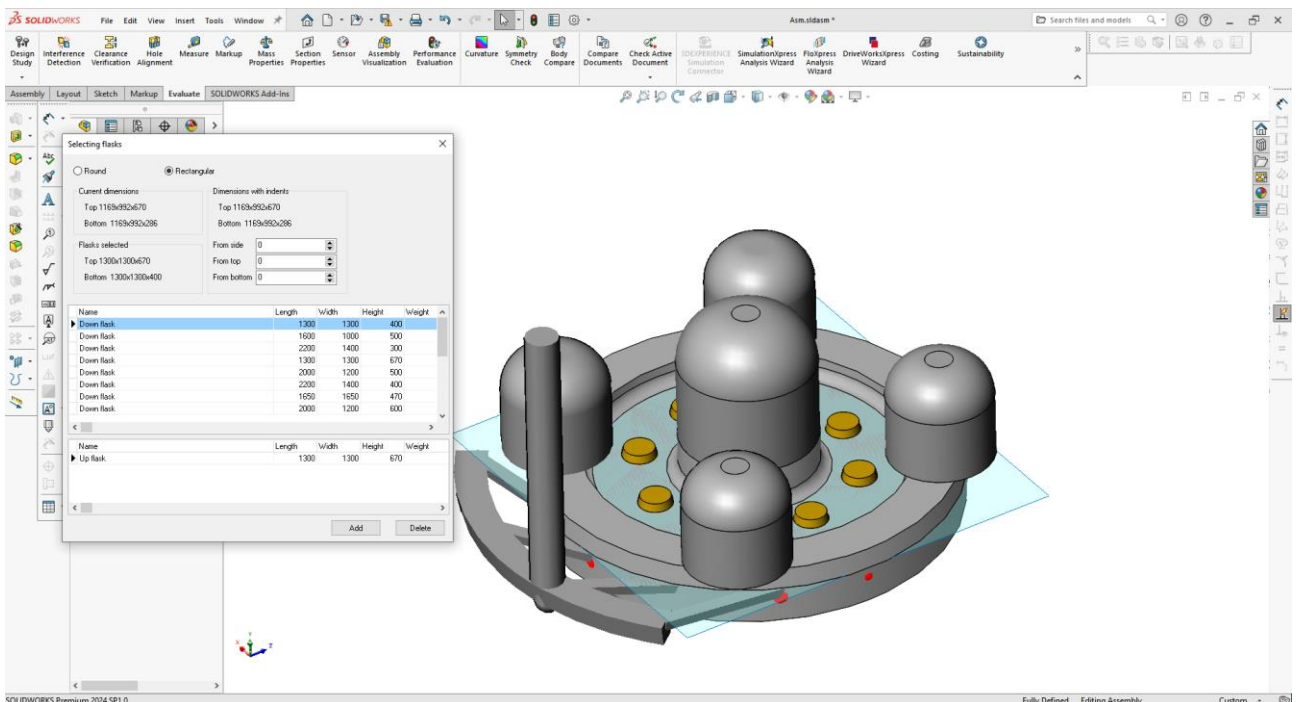


Рис. 2.13. Діалог вибору опок зі списку підходящих за розміром

Результат операції «Додати опоки» наведено на рис. 2.14.

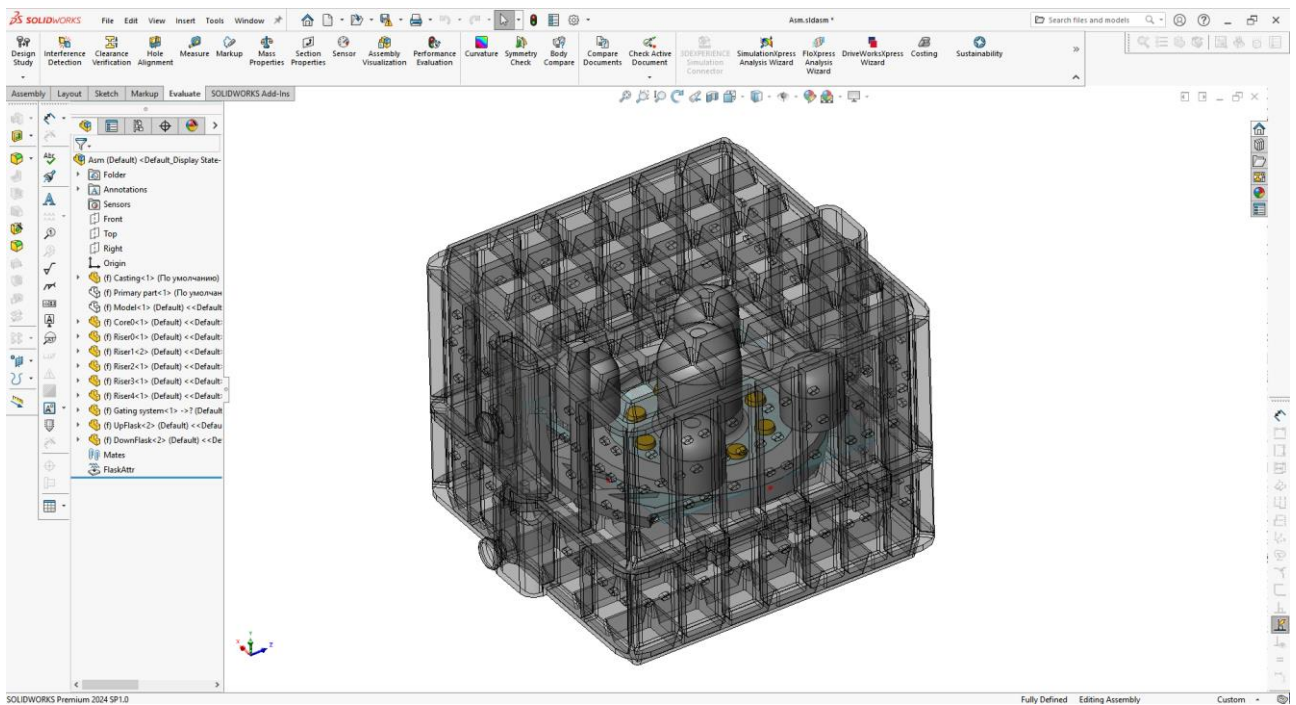


Рис. 2.14. Результат операції «Додати опоки»

Для вставки півформ у 3D модель технології викликається команда «Сформувати» (рис. 2.15).

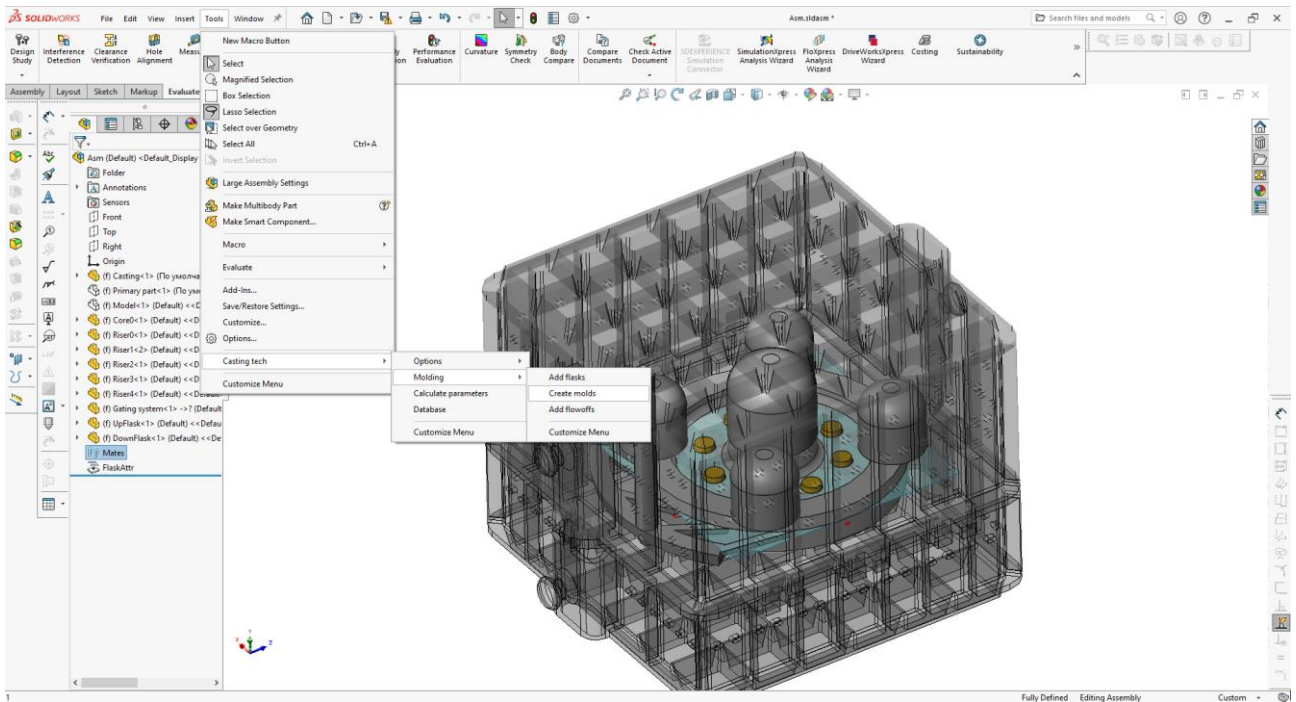


Рис. 2.15. Команда меню «Сформувати»

Після виклику цієї команди користувачу відображається діалог вибору матеріалів моделі, форми, покриття та режимів сушіння (рис. 2.16).

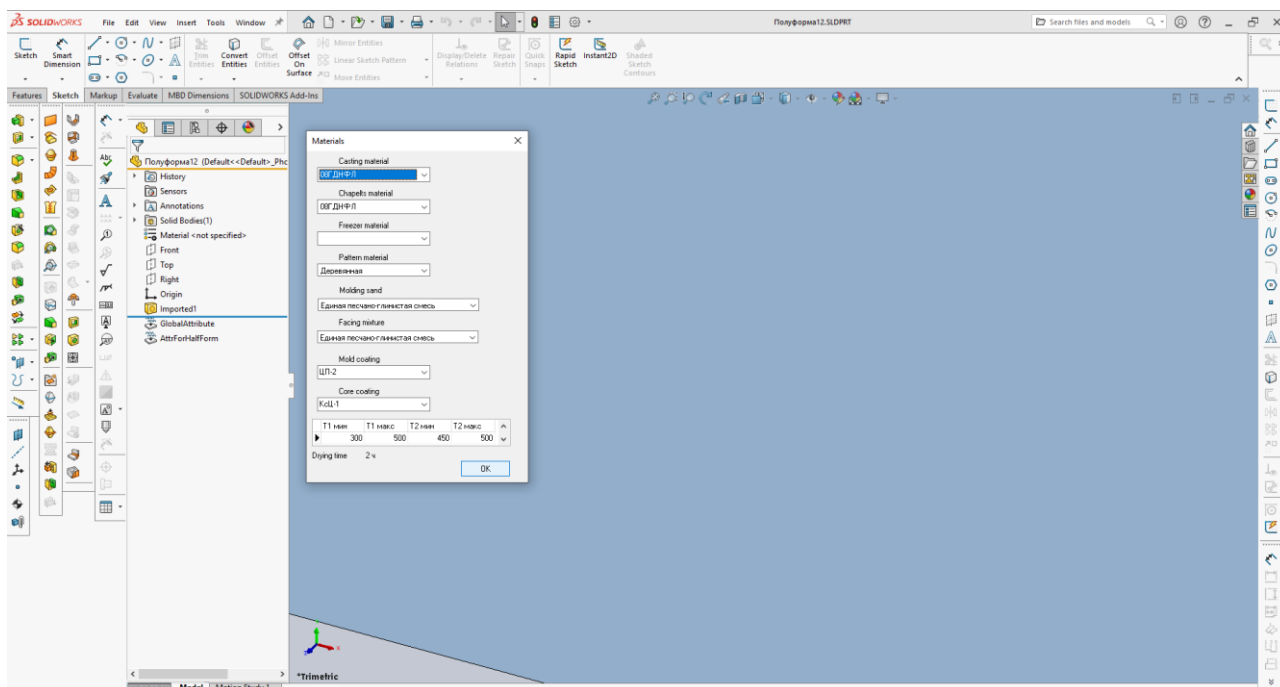


Рис. 2.16. Діалог вибору матеріалів моделі, форми, покриття та режимів сушіння

Після здійснення користувачем вибору матеріалів та натискання кнопки ОК формуються 3D моделі півформ і вставляються у 3D модель технології (рис. 2.17).

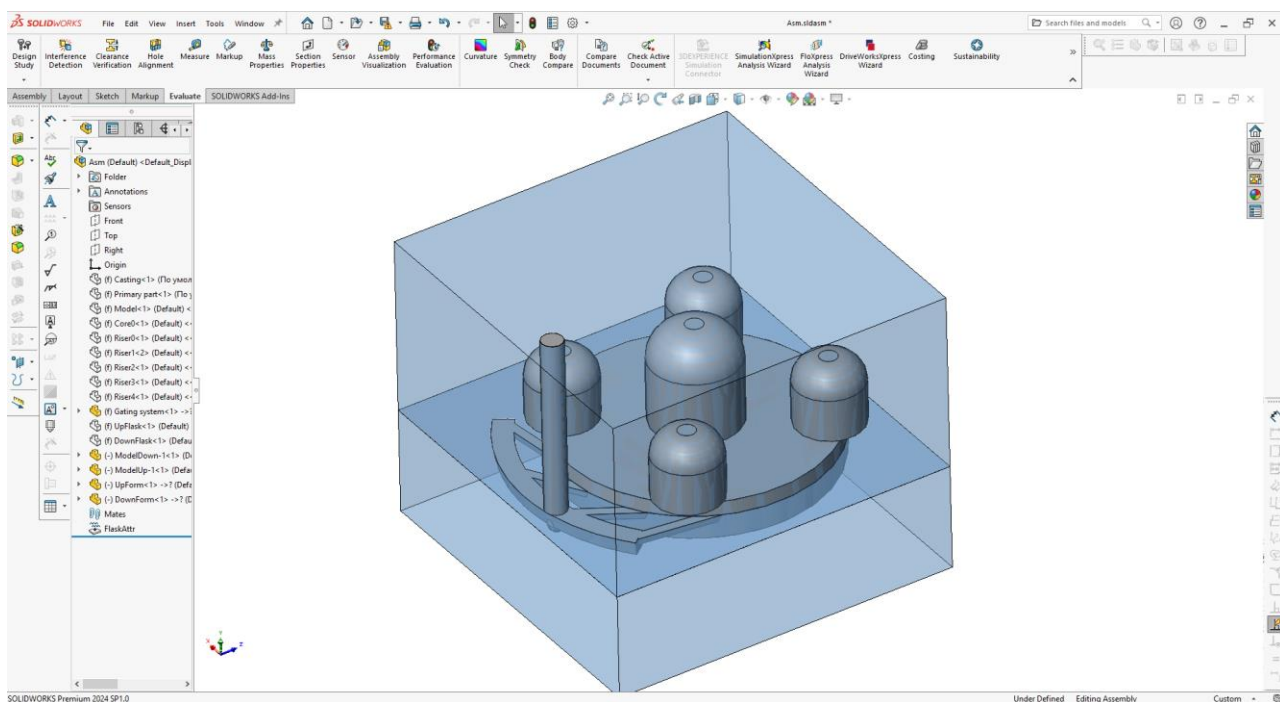


Рис. 2.17. 3D модель технології з вставленими моделями півформ та півмоделей

На рис. 2.18 наведено результат операції «Сформувати» – за результатами розрахунку розмірів створено та вставлено в 3D модель технології 3D моделі півформ з відповідними порожнинами.

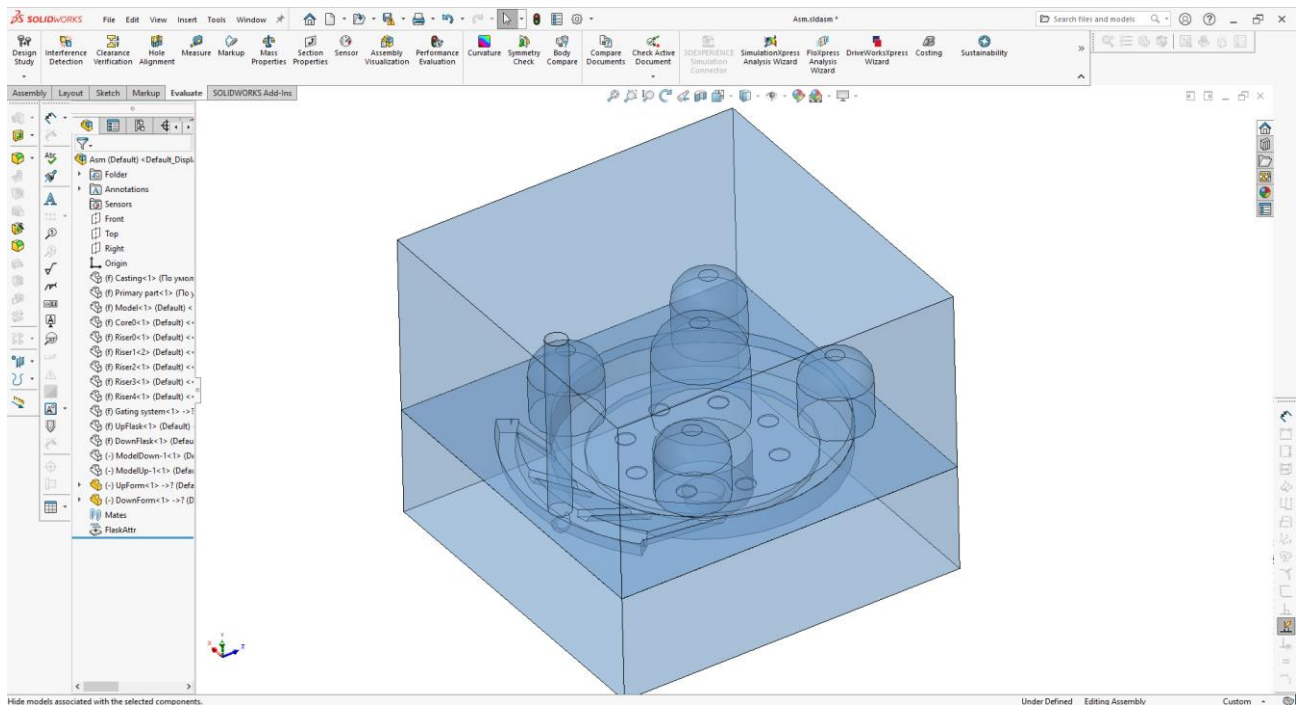


Рис. 2.18. Порожнини в 3D моделях півформ

Окремо 3D модель верхньої півформи наведено на рис. 2.19.

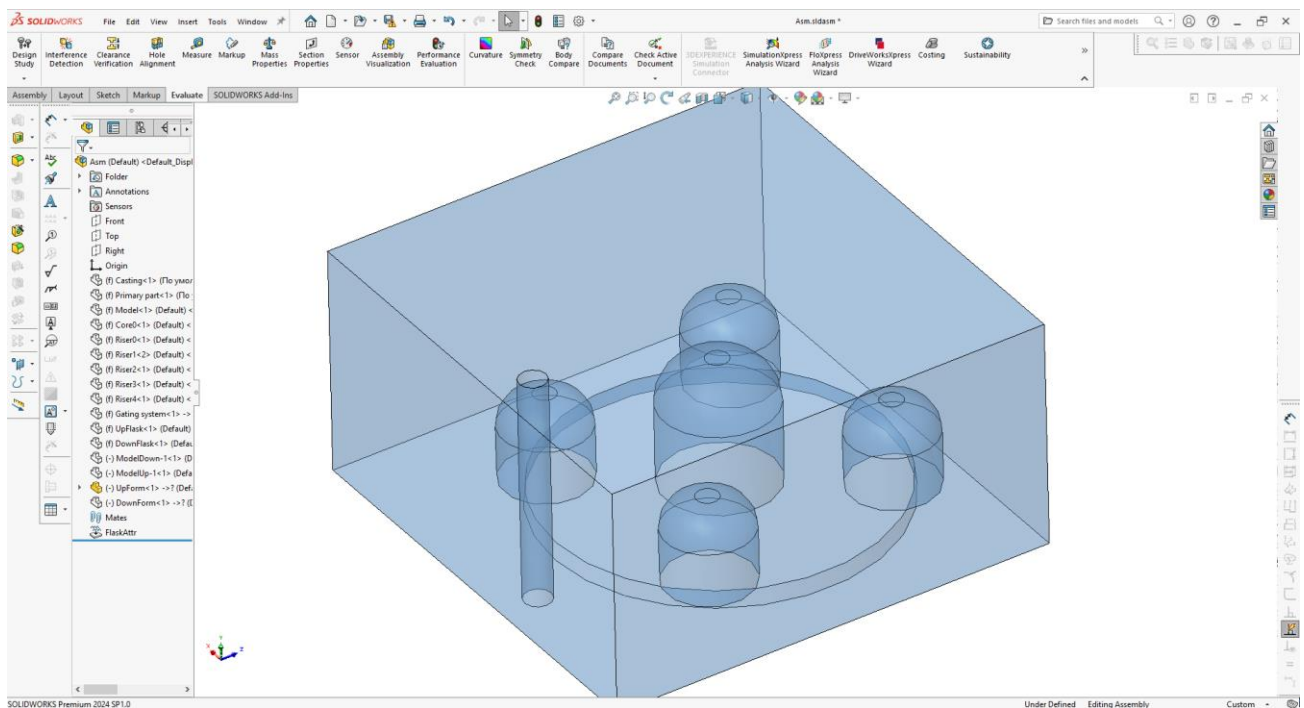


Рис. 2.19. 3D модель верхньої півформи

Окремо 3D модель нижньої пів форми наведено на рис. 2.20.

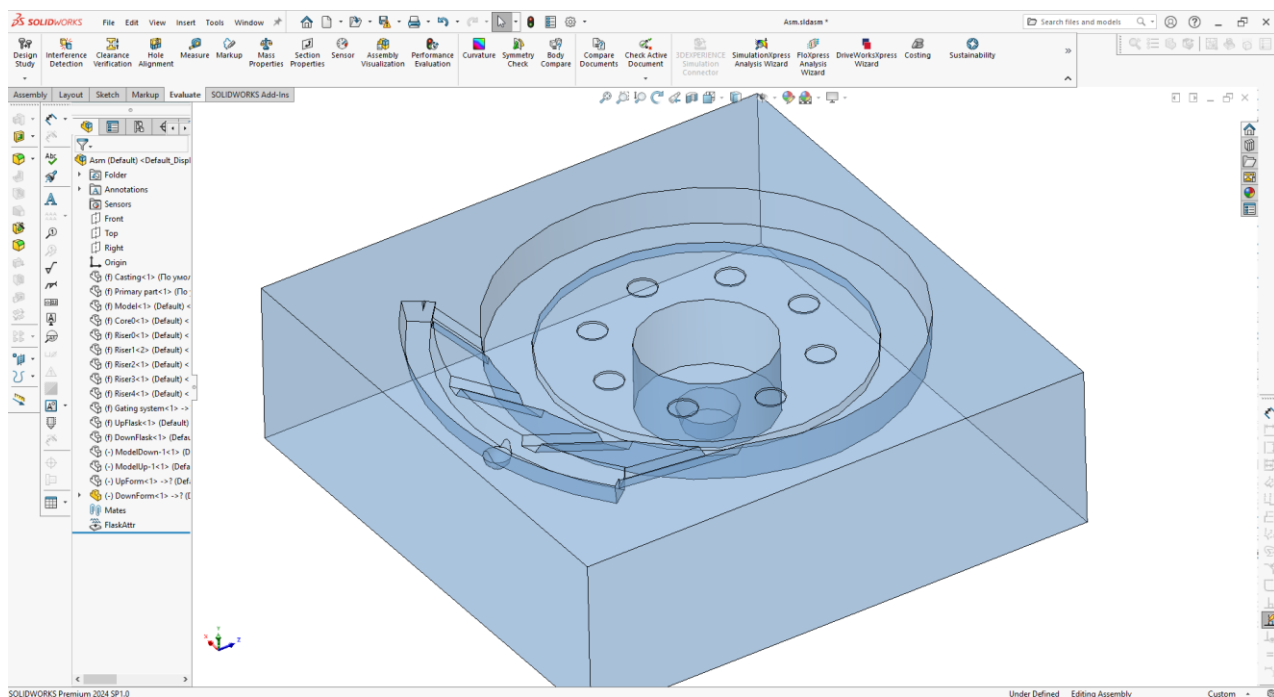


Рис. 2.20. 3D модель нижньої півформи

Також при виконанні команди «Сформувати» створюються і вставляються в 3D модель технології моделі верхньої та нижньої формувальних півмоделей (рис. 2.21, 2.22), які можуть бути використані для їх виготовлення вручну або на верстаті з ЧПУ.

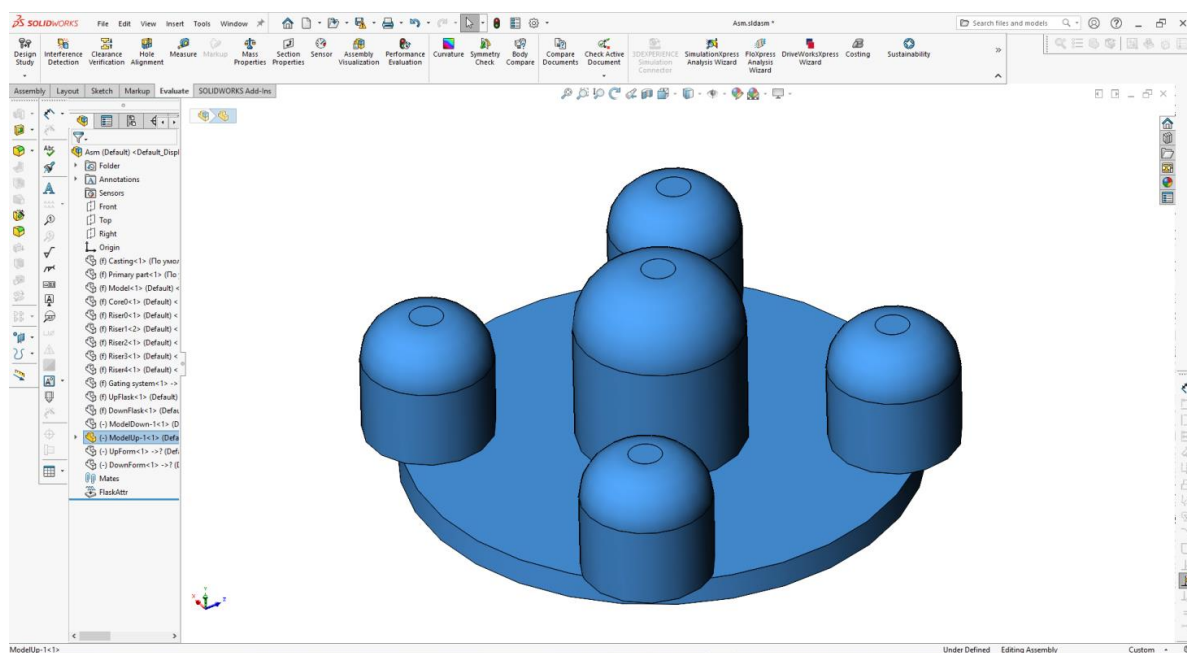


Рис. 2.21. 3D модель верхньої формувальної півмоделі

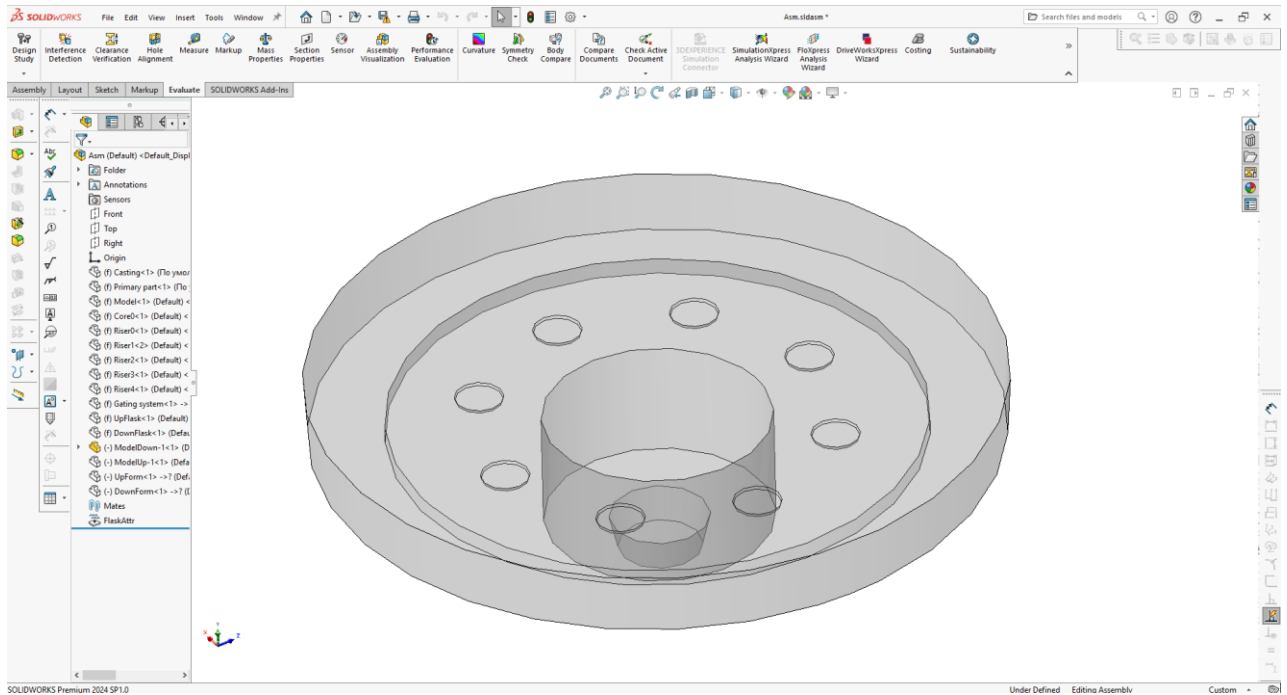


Рис. 2.22. 3D модель нижньої формувальної півмоделі

Для додавання випорів викликається команда «Додати випори» (рис. 2.23).

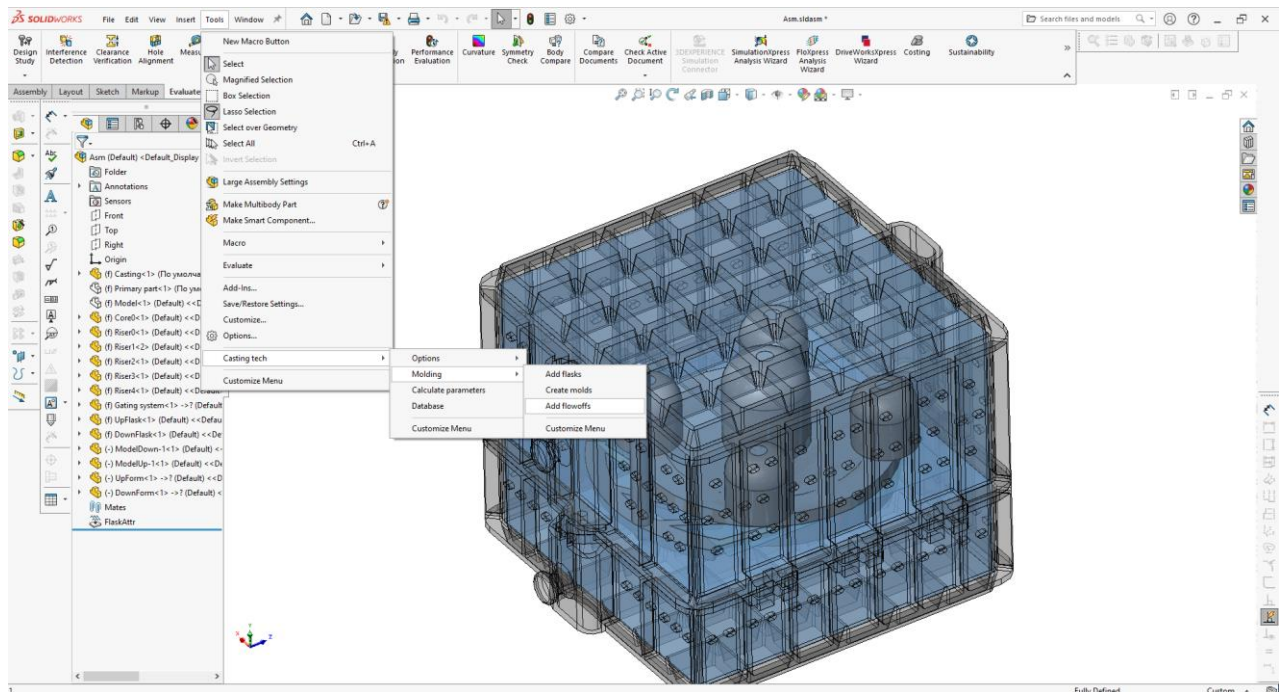


Рис. 2.23. Команда меню «Додати випори»

Після виклику цієї команди користувачеві відображається діалог створення та додавання випорів. У цьому діалозі необхідно вказати метод розрахунку

випору – використовувати радіуси або кут нахилу. І ввести константи у відповідні поля діалогу залежно від вибраного методу (рис. 2.24).

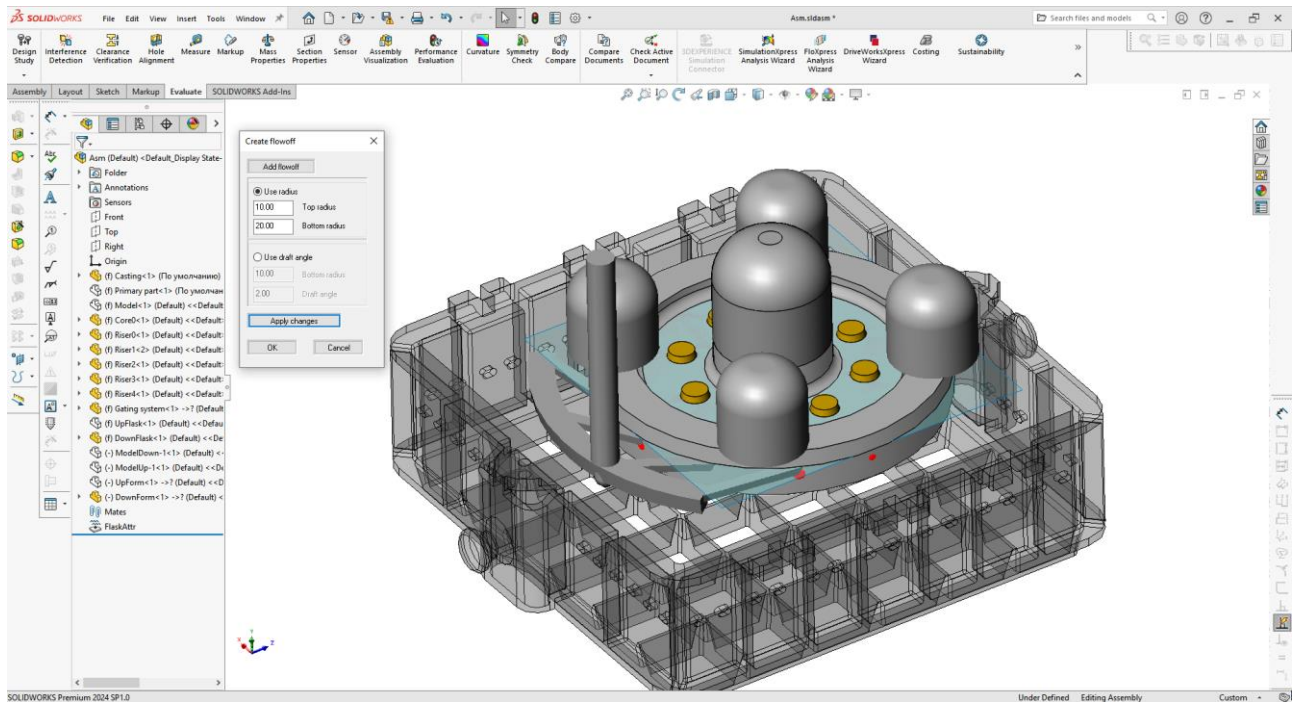


Рис. 2.24. Діалог створення та додавання випорів

Після вибору параметрів натиснути кнопку «Додати випор» та вказати на 3D моделі точку для вставки випору. При цьому автоматично проводиться розрахунок розмірів випору в залежності від параметрів вказаної точки, створюється 3D модель випору відповідно до розрахованих розмірів та вставляється у вказану точку на 3D моделі технології (рис. 2.25, 2.26).

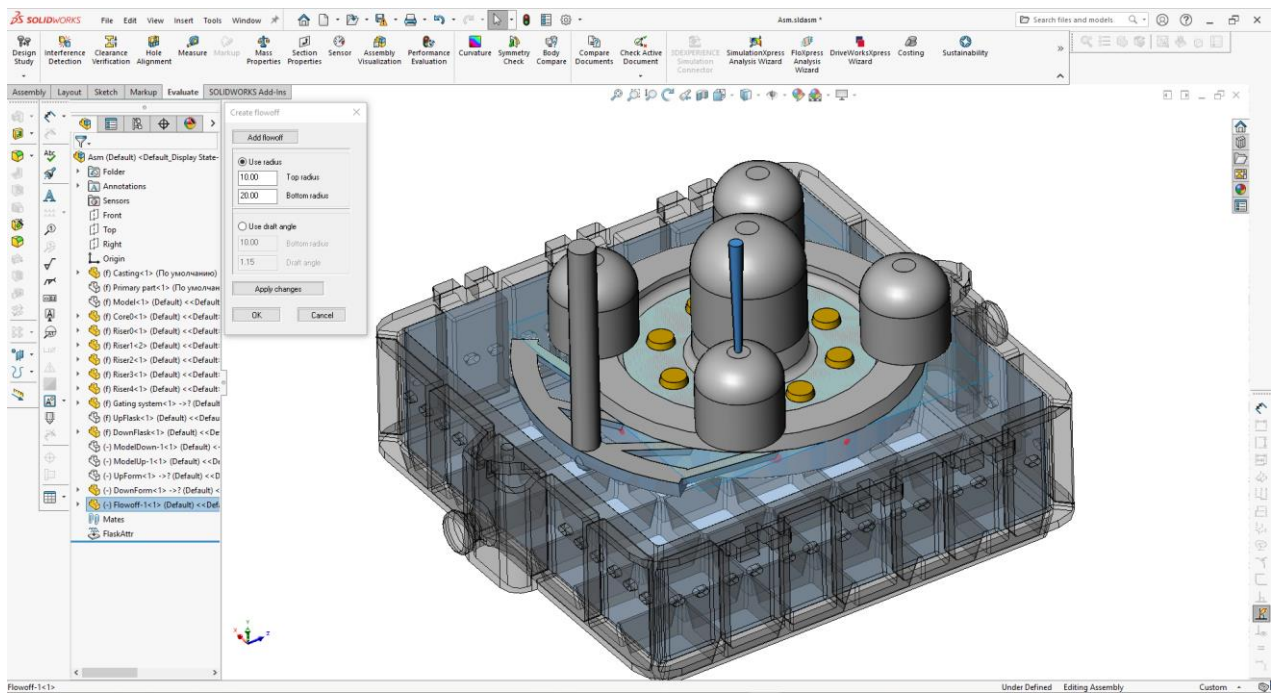


Рис. 2.25. Створення та вставка 3D моделі довгого випору у вказану точку на 3D моделі технології

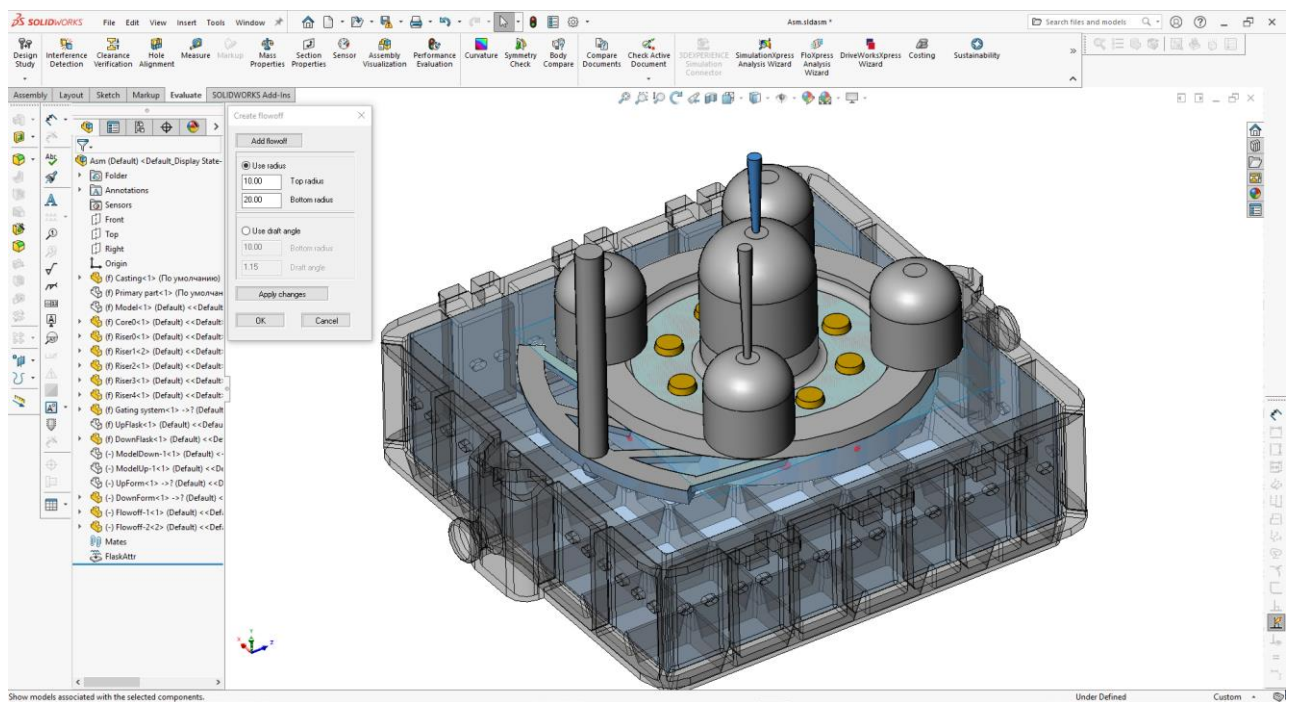


Рис. 2.26. Створення та вставка 3D моделі короткого випору у вказану точку на 3D моделі технології

Для редагування списку опок, їх геометричних параметрів, редагування додаткових параметрів формування, а також управління зберіганням 3D моделей опок викликається команда «База даних» (рис. 2.27).

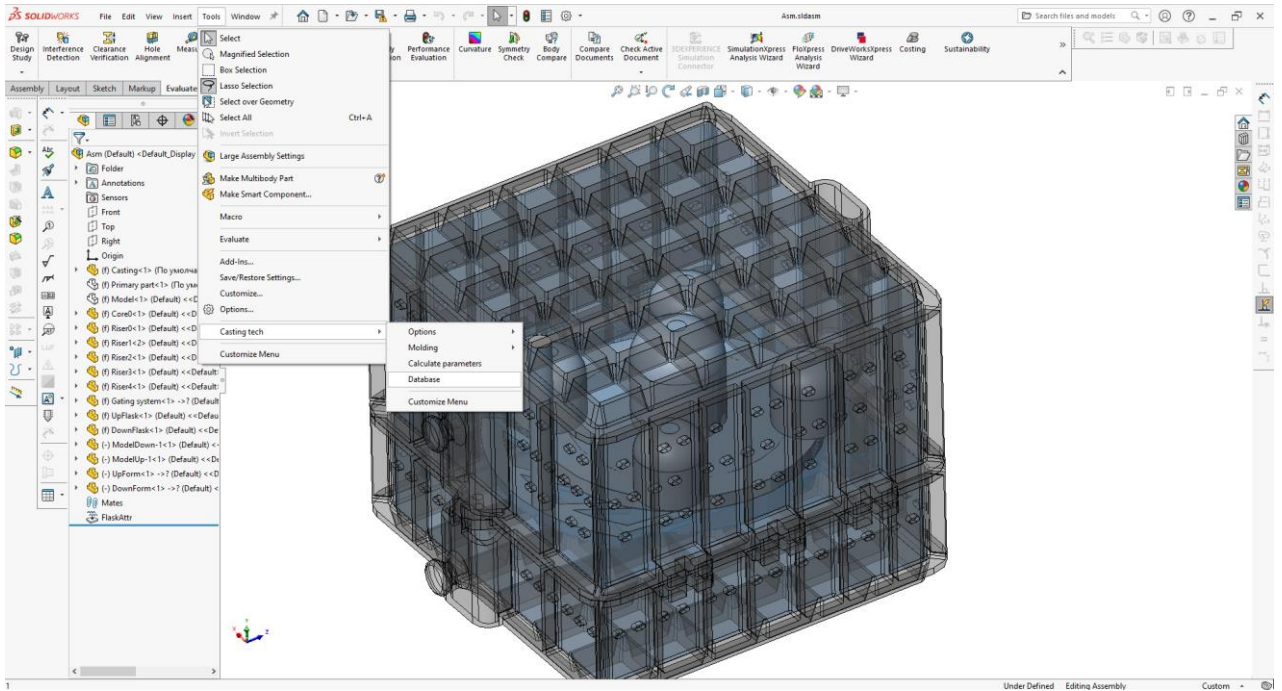


Рис. 2.27. Команда меню «База даних»

Після виклику цієї команди з'являється діалогове вікно з деревоподібним списком таблиць, доступних для редагування (рис. 2.28 – 2.30).

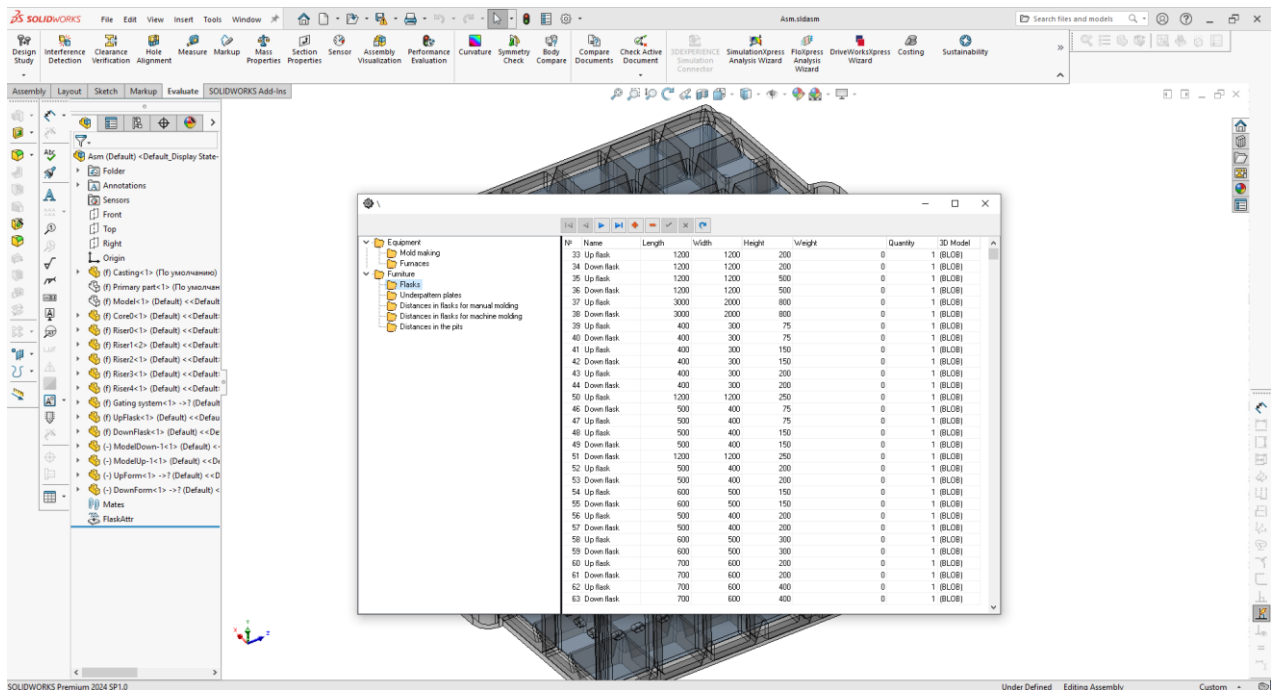


Рис. 2.28. Таблиця бази даних з параметрами опок

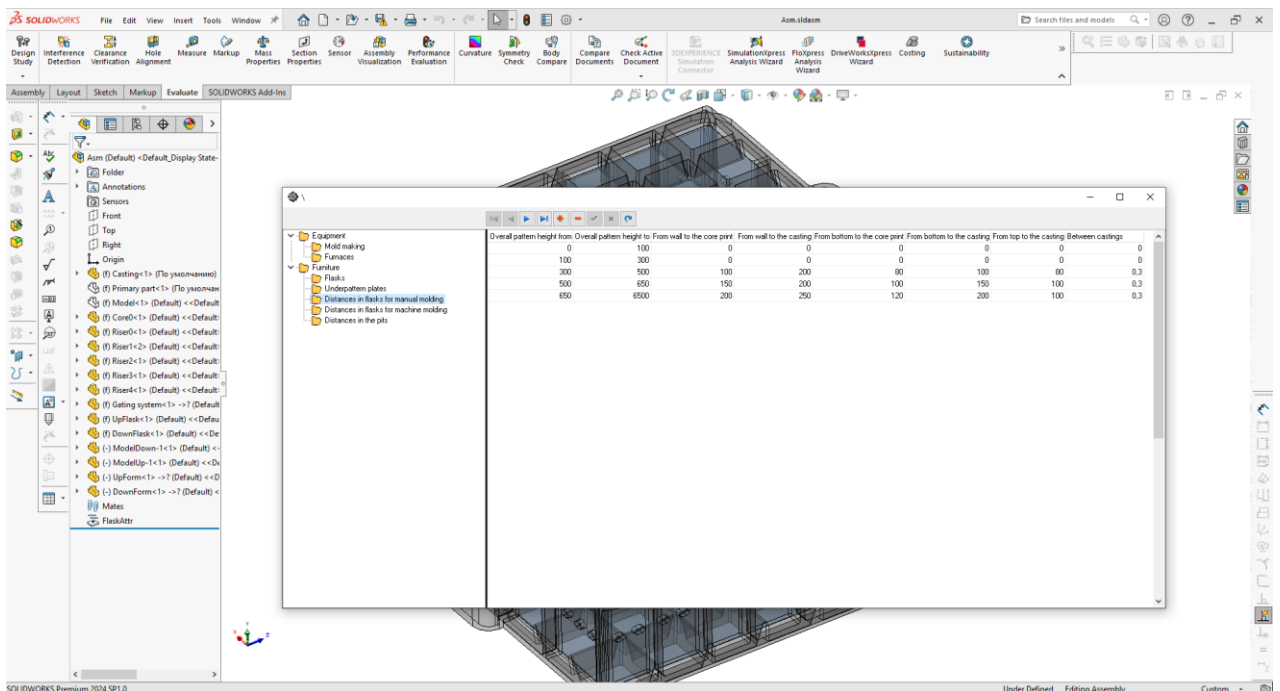


Рис. 2.29. Таблица бази даних з параметрами відступів при ручному формуванні

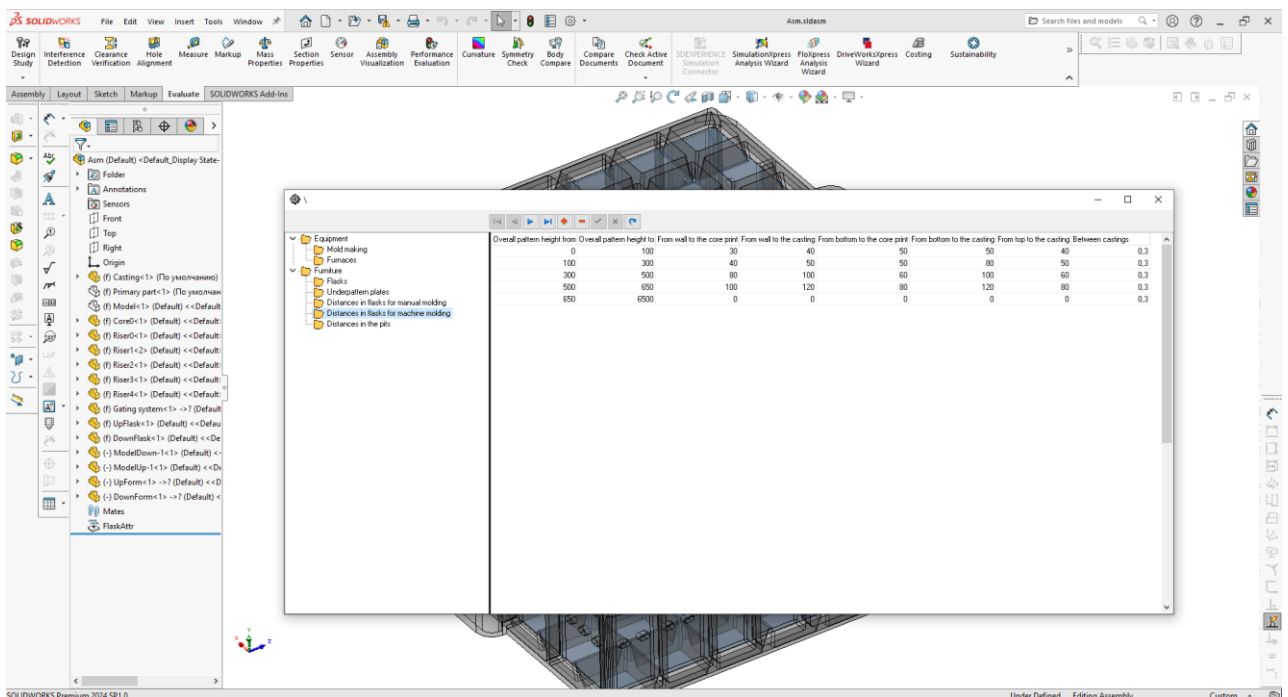


Рис. 2.30. Таблица бази даних з параметрами відступів при машинному формуванні

Для розрахунку та виведення основних результуючих технологічних параметрів (загальна маса ливарної збірки, маса суміші, маса і об'єм покриття тощо) викликається команда «Результат розрахунків» (рис. 2.31).

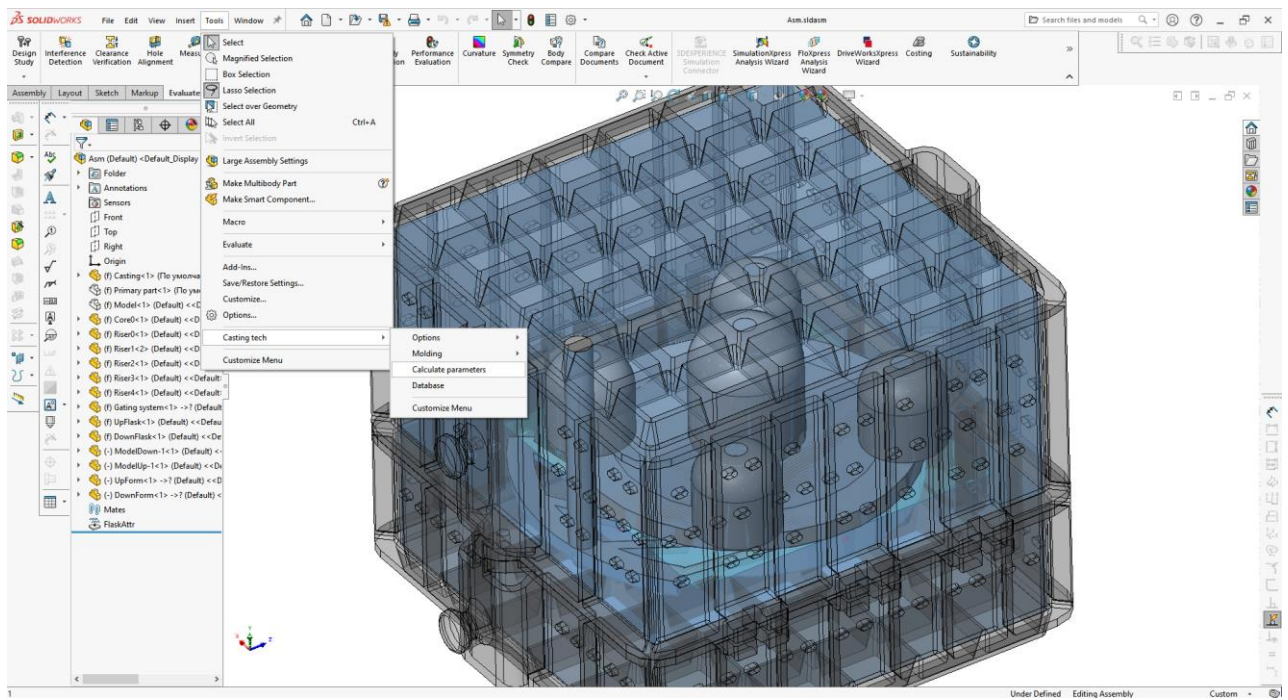


Рис. 2.31. Команда меню «Результат розрахунків»

Створена програма надає користувачу можливість легше проводити стандартизацію й уніфікацію елементів технологічного оснащення та систематизацію проєктних рішень, які дають змогу скоротити терміни технологічної підготовки виробництва відливків, зменшити собівартість продукції ливарного виробництва, підвищити якість і конкурентоспроможність виробів, що випускаються.

ВИСНОВКИ

Метою виконання кваліфікаційної роботи є розробка інженерного плагіну для програмного комплексу САПР SolidWorks.

У рамках цієї роботи розроблено автоматизовані функції, пов'язані з розрахунком та отриманням 3D моделей, що стосуються формувальних операцій процесу лиття. Практичне значення отриманих результатів полягає у створенні елементів автоматизованого проєктування певних етапів ливарної технології за допомогою вбудованого в систему SolidWorks API з метою максимального спрощення та прискорення певних етапів проєктування.

Функції API надають програмістам прямий доступ до функцій SolidWorks. API містить сотні функцій, які можна викликати з файлів макросів Visual Basic (VB), Visual Basic for Applications (VBA), VB.NET, C++, C# або SolidWorks. Використання API для автоматизації SolidWorks виконується через розробку Add-in додатків.

Розроблена система написана мовою C++ стандарту C++17 та складається з двох модулів – модулю плагіна SolidWorks та модулю бази даних. Так, плагін SolidWorks реалізує роботу команд, пов'язаних зі зчитуванням і зміною 3D моделей, використовуючи API SolidWorks. Модуль бази даних забезпечує зберігання та вибірку даних, необхідних для побудови елементів 3D моделей (розміри, константи для розрахунків, матеріали тощо), а також зберігання та видачу 3D моделей стандартних елементів технології (опоки, підмодельні плити). Модуль бази даних та діалоги користувача розроблені у вигляді окремої DLL.

Firebird надає клієнтські бібліотеки для взаємодії із сервером баз даних. Ці бібліотеки є доступними для різних мов програмування, таких як Java, C++, Python та інших, що забезпечує широку підтримку для розробників.

В роботі наведено алгоритми команд (функцій) «Додати опоки», «Сформувати», «Додати випори» модуля SolidWorks.

Вихідними даними програми є модифікована 3D модель ливарної технології з доданими 3D моделями опок, півформ, формувальних моделей і випорів, а також результат розрахунку відповідних технологічних параметрів – загальної маси ливарної збірки, маси суміші, маси та об’єму покриття.

До переваг даної розробки можна віднести:

- високу точність розрахунків і моделей;
- зменшення часу на проектування;
- інтеграцію з вже існуючими системами SolidWorks;
- зручний та інтуїтивно зрозумілий інтерфейс користувача;
- можливість масштабування та адаптації під різні потреби виробництва.

Створена програма надає користувачу можливість легше проводити стандартизацію й уніфікацію елементів технологічного оснащення та систематизацію проектних рішень, які дають змогу скоротити терміни технологічної підготовки виробництва відливків, зменшити собівартість продукції ливарного виробництва, підвищити якість і конкурентоспроможність виробів, що випускаються.

Визначено трудомісткість розробленого інженерного плагіну для програмного комплексу САПР SolidWorks (630,23 людино-годин), проведений підрахунок вартості роботи по створенню програми (208314,85 грн.) та розраховано час на його створення (3,6 місяці).