

УДК 621

О.Л. Ляшук, д.т.н., проф.; О.П. Цьонь, к.т.н., доц.; С.М. Паньків, О.А. Юр'єв
Тернопільський національний технічний університет ім. І. Пулюя, Україна

АЛГОРИТМ РОБОТИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ РОЗРАХУНКУ ПРОДУКТИВНОСТІ ГВИНТОВОГО КОНВЕЄРА

O.L. Lyashuk, Dr. Sci., Prof; O.P. Tson, Ph.D., Assoc. Prof.; S.M. Pankiv, O.A. Yuriev
SOFTWARE WORKING ALGORITHM FOR CALCULATING SCREW
CONVEYOR PRODUCTIVITY

Для визначення продуктивності гвинтового конвеєра була створена програма на мові програмування C#. На основі математичної моделі розроблено блок-схему для перевірки її адекватності (рис. 1).

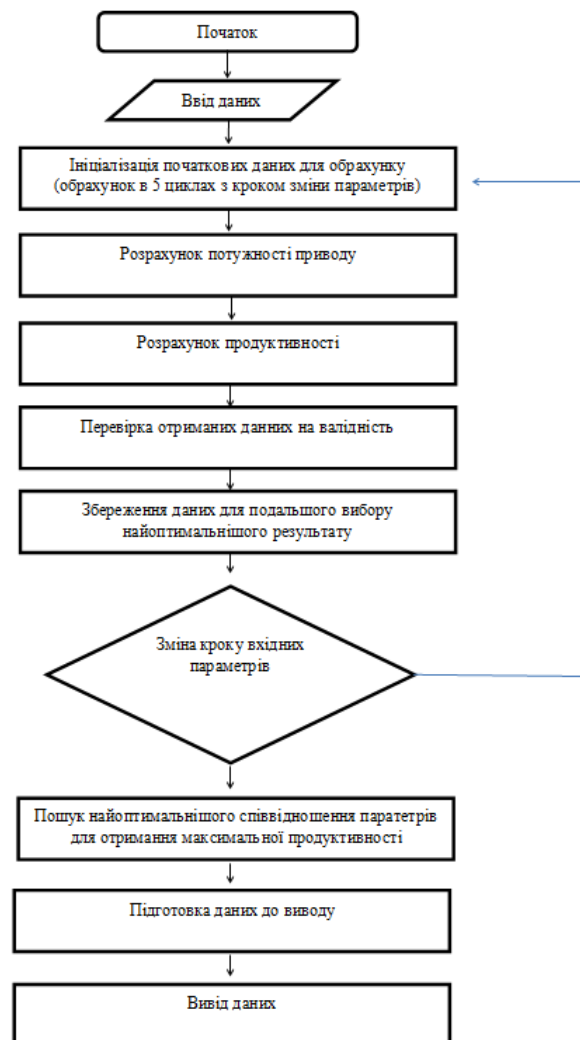


Рисунок 1. Блок-схема перевірки адекватності математичної моделі

На основі моделі було обрано найбільш значущі параметри, після чого виконано розрахунок продуктивності для кожної комбінації вхідних параметрів (рис. 2).

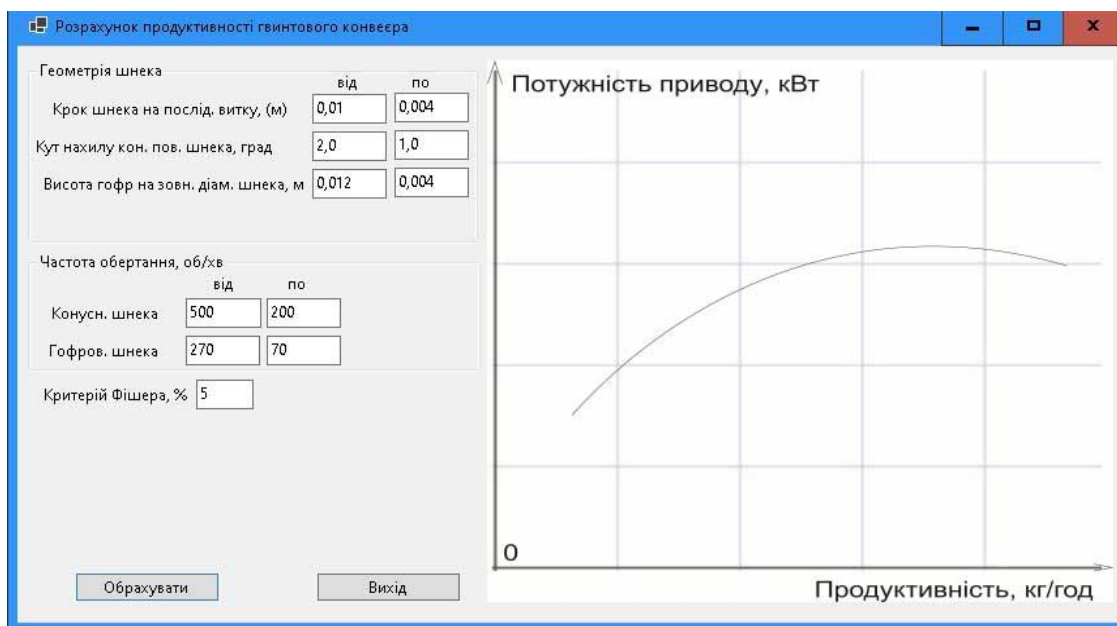


Рисунок. 2. Вікно програми розрахунку продуктивності

Алгоритм роботи програмного забезпечення:

1. Задається діапазон допустимих значень для кожного набору вхідних даних і визначається крок обчислень.
 2. Розраховується потужність на основі фіксованих значень вхідних параметрів.
 3. Визначається продуктивність для тих самих фіксованих значень вхідних параметрів.
 4. Перевіряються отримані результати на валідність.
 5. Збереження отриманих даних для подальшого аналізу.
 6. Зміна кроку вхідного параметра і повторення обчислень.
 7. Після завершення всіх обчислень отримується продуктивність для кожного набору вхідних параметрів.
 8. Вибір найоптимальнішого результату серед усіх наборів вхідних даних.
- З огляду на наявність п'яти основних параметрів і необхідність проведення обчислень для всіх можливих комбінацій вхідних даних, було вирішено розподілити розрахунки на окремі потоки. У програмі використано наступні технології:
1. Parallel Tasks – забезпечило паралельне виконання розрахунків, що значно підвищило швидкість роботи.
 2. Рекурсія – розроблені рекурсивні процедури для пошуку розв'язків.
 3. LINQ – пошук оптимального результату здійснювався за допомогою LINQ, при цьому проміжні результати зберігалися в п'ятивимірному масиві, що відповідає кількості вхідних параметрів.
 4. ADO.NET – проміжні результати обчислень зберігалися в таблицях бази даних SQL Server.
 5. DynamicDataDisplay – бібліотека з відкритим кодом, яка використовувалася для побудови графіків.
 6. Linear Programming – використана бібліотека для проведення математичних обчислень.
- Програма дозволяє зберігати всі проміжні результати розрахунків, які можуть бути використані для додаткового аналізу. Через складність математичної моделі було виконано додаткову валідацію проміжних результатів для забезпечення їх коректності.