АНОТАЦІЯ

Розроблене програмне забезпечення призначене для оптимізації режимів токарної обробки титану з субмікро- та нанокристалічною структурою за допомогою генетичного алгоритму. Оптимізуються функцій продуктивності



та розміру зерна металу



завдяки чому обробка металу проводиться з максимальною продуктивністю та з мінімальним ростом зерна металу, що запобігає втраті нанокристалічної структури.

Багатокритеріальна оптимізаця виконується завдяки методу Шаффера.

Програма виконує усі етапи роботи генетичного алгоритму:

* ініціалізація генетичного алгоритму;
* оцінка пристосованості популяції;
* перевірка умови зупинки алгоритму;
* селекція найкращих особин популяції;
* застосування генетичних операторів;
* формування нової популяції;
* вибір найкращого результату.

Web-додаток розроблений на ПЕОМ мовами PHP, HTML, з використанням каскадних таблиць стилю CSS. Програмне забезпечення розгорнуте на сервері Apache.

Вимоги до апаратного забезпечення: додаток повинен бути розгорнутий на веб-сервері Apache. Для доступу до програми необхідна ПЕОМ на базі процесора Intel Pentium 4 і вище з ОЗУ не менше 512МБ, операційна система MS Windows XP і вище з доступом до сеті «Інтернет».

Сумарній об’єм проекту програми 1 МБ. Результатом роботи програми є виведення результатів роботи генетичного алгоритму на екран.

Вхідними даними для роботи web-додатку є введені користувачем параметри для роботи генетичного алгоритму – точність підрахунку для кожного параметру, та кількість особин у популяції.

Результатом роботи програми є вивід результатів розрахунків на екран.

# Зміст

|  |  |
| --- | --- |
| 1. Загальні відомості……………………………………………….............. | 5 |
| 1. Функціональне призначення……………………………………………. | 6 |
| 1. Опис логічної структури………………………………………………... | 9 |
| 1. Використовувані технічні засоби………………………………............. | 21 |
| 1. Виклик и завантаження……………………………..…………………... | 22 |
| 1. Вхідні дані………………………………………………………………... | 23 |
| 1. Вихідні дані………………………………………………………………. | 24 |
| 1. Перелік використаних джерел інформації……………………………... | 25 |

# ЗАГАЛЬНІ ВІДОМОСТІ

Назва розробленого web-додатку – «Genetic». Розроблене програмне забезпечення призначене для оптимізації режимів токарної обробки титану з субмікро- та нанокристалічною структурою за допомогою генетичного алгоритму. Програма орієнтована на використання на HTTP-сервері Apache (кроссплатформенне ПО, підтримує операційні системи Linux, BSD, Mac OS, Microsoft Windows, Novell NetWare, BeOS). Програмне забезпечення написане з використанням HTML, PHP. Середовище розробки – набір дистрибутивів та програмна оболонка для створення web-додатків Denwer, блокнот Notepad++.

1. ФУНКЦІОНАЛЬНЕ ПРИЗНАЧЕННЯ

Дослідження провідних вчених у галузі металознавства показали, що в субмікро- та нанокристалічних металах фізико-механічні властивості суттєво перевищують відповідні показники у крупнокристалічних металах. Межа плинності збільшується до 1,5 разу, а мікротвердість – до 2 разів. В даний час розроблені ефективні технології інтенсивної пластичної деформації (кручення під високим тиском, рівноканальне кутове пресування, всебічна ковка) для створення металів з субмікро- та нанокристалічною структурою. Під час інтенсивної пластичної деформації відбувається перетворення мікроструктури металу у субмікро- та нанокристалічний стан, супроводжуваний змінами кристалічної решітки та утворенням нервіноважного стану зерна.

Заготовки отримують методами інтенсивної пластичної деформації в умовах високого тиску та умовно низьких температур. У об’ємних заготівках формується субмікро- та нанокристалічна структура, що складається з зерен, розділених переважно нерівноважними високоенергетичними границями. При впливі високих температур, стають більш інтенсивними процеси рекристалізації, бо структура субмікро- та нанокристалічних металів є термічно нестабільною.

При виготовленні виробів, як правило, застосовують механічну обробку. Механічна обробка супроводжується значними пластичними деформаціями та високою температурою різання. Відзначені умови призводять до повернення структури металу до крупнокристалічного стану та втрати вихідних високих фізико-механічних властивостей.

Програма призначена для багатокритеріальної оптимізації функцій продуктивності



та розміру зерна металу



за допомогою генетичного алгоритму, де

L – довжина оброблюваної поверхні;

d – діаметр оброблюваної заготовки;

D0 – вихідний розмір зерна у субмікро- та нанокристалічному металі;

s – подача;

v – швидкість різання;

A – показник, який враховує властивості оброблюваного металу.

Для кожної з величин користувач повинен ввести коефіцієнт точності q, що відповідає за те, з точністю до якого знаку буде отриманій результат по цій змінній, довжину гену для цієї змінної, кількість кроків, на яку буде розбита область визначення.

Також користувачем вводиться розмір популяції, що бере участь у генетичному алгоритмі.

Програма виконує усі етапи роботи генетичного алгоритму:

* генерацію початкової популяції генетичного алгоритму відповідного розміру;
* оцінку пристосованості особин популяції за допомогою фітнес-функцій;
* вибір найкращих особин для селекції та створення нової популяції;
* переведення генів особин у бінарну форму для подальшого схрещення;
* схрещення найкращих особин популяції та створення нової популяції;
* переведення генів особин нової популяції у десятичну форму для подальшої перевірки пристосованості популяції.

Зважаючи на те, що поставлена задача передбачає багатокритеріальну оптимізацію параметрів по декільком функціям, для цього був використаний метод незалежноі селекції Шаффера, де для кожної функції береться окрема під популяція, але схрещення відбувається для усієї загальної популяції.

Функціональні обмеження – web-додаток повинен бути розгорнутий на сервері. Для роботи з додатком необхідна ПЕОМ з доступом до сіті «Інтернет» та будь-яким сучасним браузером.

3 ОПИС ЛОГІЧНОЇ СТРУКТУРИ

Веб-додаток призначений для оптимізації режимів токарної обробки титану з субмікро- та нанокристалічною структурою за допомогою генетичних алгоритмів. Цільовими функціями виступають функцій продуктивності



та розміру зерна металу



де

L – довжина оброблюваної поверхні;

d – діаметр оброблюваної заготовки;

D0 – вихідний розмір зерна у субмікро- та нанокристалічному металі;

s – подача;

v – швидкість різання;

A – показник, який враховує властивості оброблюваного металу.

Генетичні алгоритми виникли в результаті спостереження і спроб копіювання природних процесів, що відбуваються в світі живих організмів, зокрема, еволюції та пов'язаної з нею селекції (природного відбору) популяцій живих істот.

Ідею генетичних алгоритмів висловив Дж. Холланд у кінці шістдесятих — початку сімдесятих років XX століття. Він зацікавився властивостями процесів природної еволюції (в тому числі фактом, що еволюціонують хромосоми, а не самі живі істоти). Холланд був упевнений у можливості скласти і реалізувати у вигляді комп'ютерної програми алгоритм, який буде вирішувати складні задачі так, як це робить природа — шляхом еволюції.

Тому він почав працювати над алгоритмами, що оперували послідовностями двійкових цифр (одиниць і нулів), що одержали назву хромосом. Ці алгоритми імітували еволюційні процеси в поколіннях таких хромосом. У них були реалізовані механізми селекції та репродукції, аналогічно вживаними при природній еволюції.

Так само, як і в природі, **генетичні алгоритми** здійснювали пошук «хороших» хромосом без використання будь-якої інформації про характер розв'язуваної задачі. Була потрібна тільки якась оцінка кожної хромосоми, яка відображає її пристосованість. Механізм селекції полягає у виборі хромосом з найвищою оцінкою (тобто найбільш пристосованих), які репродукують частіше, ніж особини з більш низькою оцінкою (гірше пристосовані).

Репродукція означає створення нових хромосом у результаті рекомбінації генів батьківських хромосом. Рекомбінація — це процес, в результаті якого виникають нові комбінації генів. Для цього використовуються дві операції: схрещування, що дозволяє створити дві зовсім нові хромосоми нащадків шляхом комбінування генетичного матеріалу пари батьків, а також мутація, яка може викликати зміни в окремих хромосомах.

У генетичних алгоритмах застосовується ряд термінів, запозичених з генетики, перш за все гени і хромосоми, а також популяція, особина, алель, генотип, фенотип.

**Генетичні алгоритми** застосовуються при розробці програмного забезпечення, в системах штучного інтелекту, оптимізації, штучних нейронних мережах і в інших галузях знань. Слід зазначити, що з їх допомогою вирішуються завдання, для яких раніше використовувалися тільки нейронні мережі. У цьому випадку генетичні алгоритми виступають просто в ролі незалежного від нейронних мереж альтернативного методу, призначеного для вирішення тієї ж самої задачі.

**Генетичний алгоритм** являє собою метод, що відображає природну еволюцію методів вирішення проблем, і в першу чергу задач оптимізації. Генетичні алгоритми — це процедури пошуку, засновані на механізмах природного відбору і спадкоємства. У них використовується еволюційний принцип виживання найбільш пристосованих особин. Вони відрізняються від традиційних методів оптимізації декількома базовими елементами. Зокрема, генетичні алгоритми:

* обробляють не значення параметрів самого завдання, а їх закодовану форму;
* здійснюють пошук рішення виходячи не з єдиної точки, а з їх деякої популяції;
* використовують тільки цільову функцію, а не її похідні або іншу додаткову інформацію;
* застосовують імовірнісні, а не детерміновані правила вибору.

Перераховані чотири властивості, які можна сформулювати також як кодування параметрів, операції на популяціях, використання мінімуму інформації про завдання і рандомізація операцій приводять у результаті до стійкості генетичних алгоритмів і до їх переваги над іншими широко вживаними технологіями.

При описі генетичних алгоритмів використовуються визначення, запозичені з генетики. Наприклад, мова йде про популяцію особин, а в якості базових понять застосовуються ген, хромосома, генотип, фенотип, алель. Також використовуються відповідні цим термінам визначення з технічного лексикону, зокрема, ланцюг, двійкова послідовність, структура.

**Популяція** — це кінцева множина особин.

**Особини**, що входять в популяцію, у генетичних алгоритмах представляються хромосомами з закодованими в них множинами параметрів задачі, тобто рішень, які інакше називаються точками в просторі пошуку (search points). У деяких роботах особини називаються організмами.

**Хромосоми** (інші назви — ланцюжки або кодові послідовності) — це впорядковані послідовності генів.

**Ген** (який також називається властивістю, знаком чи детектором) — це атомарний елемент генотипу, зокрема, хромосоми.

**Генотип** або структура — це набір хромосом даної особини. Отже, особинами популяції можуть бути генотипи або одиничні хромосоми (в досить поширеному випадку, коли генотип складається з однієї хромосоми).

**Фенотип** — це набір значень, які відповідає даному генотипу, тобто декодована структура або безліч параметрів задачі (розв’язок, точка простору пошуку).

**Алель** — це значення конкретного гена, також визначається як значення властивості або варіант властивості.

**Локус** чи позиція вказує місце розміщення даного гена в хромосомі (ланцюжку).

Дуже важливим поняттям у генетичних алгоритмах вважається **функція пристосованості** (fitness function), яка інакше називається функцією оцінки. Вона являє міру пристосованості даної особини в популяції. Ця функція відіграє найважливішу роль, оскільки дозволяє оцінити ступінь пристосованості конкретних особин у популяції і вибрати з них найбільш пристосовані (тобто мають найбільші значення функції пристосованості) відповідно з еволюційним принципом виживання «найсильніших» (які найкраще пристосувалися).

Функція пристосованості також отримала свою назву безпосередньо із генетики. Вона надає сильний вплив на функціонування генетичних алгоритмів і повинна мати точне і коректне визначення. У задачах оптимізації функція пристосованості, як правило, оптимізується і називається цільовою функцією.

У теорії управління функція пристосованості може приймати вигляд функції похибки, а в теорії ігор — вартісної функції.

На кожній ітерації генетичного алгоритму пристосованість кожної особини даної популяції оцінюється за допомогою функції пристосованості, і на цій основі створюється наступна популяція особин, що складають безліч потенційних рішень проблеми, наприклад, задачі оптимізації. Чергова популяція в генетичному алгоритмі називається поколінням, а до новостворюваної популяції особин застосовується термін «нове покоління» або «покоління нащадків».

Основний (класичний) генетичний алгоритм, що був використаний у розробленому програмному забезпеченні, складається з наступних кроків:

1. ініціалізація, або вибір вихідної популяції хромосом;
2. оцінка пристосованості хромосом в популяції;
3. перевірка умови зупинки алгоритму;
4. селекція хромосом;
5. застосування генетичних операторів;
6. формування нової популяції;
7. вибір «найкращої» хромосоми.

Блок — схема основного генетичного алгоритму зображена на рис. 3.1.

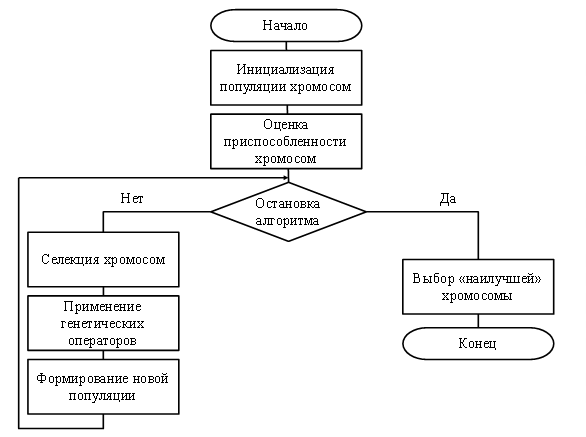


Рисунок 3.1 – Блок-схема генетичного алгоритму

Розглянемо конкретні етапи роботи додатку більш докладно.

Ініціалізація, тобто формування вихідної популяції, полягає у випадковому виборі заданої кількості особин.

На цьому кроці для кожного параметру за допомогою коефіцієнта q також визначається необхідна та достатня довжина двійкової послідовності, яка необхідна для кодування числа на інтервалі з шагом 10-q:

.

Де b – максимальне значення області, a – мінімальне значення.

Границі інтервалів для кожного параметру приведені у таблиці 3.1.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | L, м | d, м | D0, м | s, м/об | v, м/мин |
| min | 0,1 | 0,01 | 0,2\*10-7 | 1\*10-5 | 2,5 |
| max | 0,5 | 0,1 | 2,5\*10-7 | 10,23\*10-3 | 100 |

Таблиця 3.1 – границі інтервалів

Також визначається крок квантування:

Оцінювання пристосованості хромосом в популяції полягає в розрахунку функції пристосованості для кожної хромосоми цієї популяції. Для задачі мінімізації, чим менше значення цієї функції, тим вище «якість» хромосоми.

Для вирішення задачі багатокритеріальної оптимізації за допомогою генетичного алгоритму був обраний метод Шаффера. Цей метод полягає у тому, що вихідна популяція розбивається на декілька під популяцій, по кількості оптимізуємих функцій. Для кожної підгрупи окремо вираховується фітнес-функція та проводиться селекція, але схрещення для формування нової популяції відбувається між особинами обох підгруп. Цей алгоритм легко розширити для будь-якого числа критеріїв. Блок-схема алгоритму приведена на рисунку 3.2.



Рисунок 3.2 – схема методу Шаффера

У ролі фітнес-функцій у програмі виступають функції продуктивності



та розміру зерна металу



Визначення умови зупинки генетичного алгоритму залежить від його конкретного застосування. У оптимізаційних задачах, якщо відомо максимальне (або мінімальне) значення функції пристосованості, то зупинка алгоритму може відбутися після досягнення очікуваного оптимального значення, можливо — з заданою точністю.

Зупинка алгоритму також може статися у разі, коли його виконання не приводить до поліпшення вже досягнутого значення. Алгоритм може бути зупинений після закінчення певного часу виконання або після виконання заданої кількості ітерацій. Якщо умова зупинки виконана, то проводиться перехід до завершального етапу вибору «найкращої» хромосоми. В іншому випадку на наступному кроці виконується селекція.

У даній програмі умова зупинки генетичного алгоритму – зупинка покращення результату протягом п’ятдесяти ітерацій.

Селекція полягає в тому, що батьками можуть стати тільки ті особини, значення пристосованості яких не менше порогової величини, наприклад середнього значення пристосованості по популяції. Такий підхід забезпечує швидшу збіжність алгоритму. Проте, через швидку збіжність селективний вибір батьківської пари не підходить тоді, коли ставиться завдання визначення декількох екстремумів, оскільки для таких завдань алгоритм, як правило, швидко сходиться до одного з рішень. Крім того, для деяких багатовимірних завдань зі складним ландшафтом цільової функції швидка збіжність може мати невірний розв'язок. Цей недолік може бути частково компенсований використанням відповідного механізму відбору, який би «гальмував» занадто швидку збіжність алгоритму. Порогова величина в селекції може бути обчислена різними способами. Тому в літературі для генетичного алгоритму виділяють різні варіації селекції. Найбільш відомі з них — це турнірний і рулеточний(пропорційний) відбори.

Рулеточний відбір не може бути застосований у задачах мінімізації функції, тому у додатку використовується турнірний відбір. Також турнірний відбір забезпечує більш точний результат, та більшу швидкість роботи.

При турнірному відборі (tournament selection) з популяції, яка складається із N особин, вибираються випадковим чином t особин, і найкраща особина записується в проміжний масив. Ця операція повторюється N раз. Особини в отриманому проміжному масиві потім використовуються для схрещування (також випадковим чином). Розмір групи рядків, що відбираються для турніру, часто дорівнює 2. У цьому випадку говорять про двійковий (парний) турнір. Взагалі ж t називають чисельністю турніру. Перевагою даного способу є те, що він не вимагає додаткових обчислень.

Перед виконанням схрещення особин, необхідно перевести хромосоми особин у зручну для цього бінарну форму. Необхідне десятинне значення, що далі переводиться у бінарне, вираховується за формулою

де

x\* - отримане ціле десятинне значення;

a - мінімальна границя області;

hx – шаг квантування.

Далі отримане значення хромосоми переводиться у бінарну форму.

На першому етапі схрещування вибираються пари хромосом з батьківського популяції (батьківського пулу). Це тимчасова популяція, що складається з хромосом, відібраних в результаті селекції та призначених для подальших перетворень операторами схрещування і мутації з метою формування нової популяції нащадків. На даному етапі хромосоми з батьківського популяції об'єднуються в пари.

Це здійснюється випадковим способом відповідно до ймовірності схрещування Pс. Далі для кожної пари відібраних таким чином батьків розігрується позиція гена (локус) у хромосомі, що визначає так звану точку схрещування. Якщо хромосома кожного з батьків складається з L генів, то очевидно, що точка схрещування Lк представляє собою натуральне число, менше L. Тому фіксація точки схрещування зводиться до випадкового вибору числа з інтервалу [1, L-1] У результаті схрещування пари батьківських хромосом виходить така пара нащадків:

1. нащадок, хромосома якого на позиціях від 1 до Lк складається з генів першого з батьків, а на позиціях від Lк + 1 до L — із генів другого з батьків;
2. нащадок, хромосома якого на позиціях від 1 до Lк складається з генів другого з батьків, а на позиціях від Lк + 1 до L — з генів першого з батьків.

Процес кроссоверу зображений на рисунку 3.3.

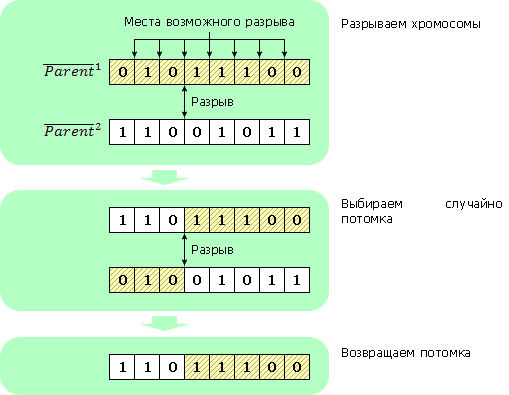


Рисунок 3.3 – схрещування

Хромосоми, отримані в результаті застосування генетичних операторів до хромосом тимчасової батьківської популяції, включаються до складу нової популяції. Вона стає так званою поточною популяцією для даної ітерації генетичного алгоритму.

Для вирахування функції пристосованості на наступній ітерації алгоритму. Необхідно повернути хромосому кожної особини у десятичну форму за допомогою формули

На кожній черговій ітерації розраховуються значення функції пристосованості для всіх хромосом цієї популяції, після чого перевіряється умова зупинки алгоритму і або фіксується результат у вигляді хромосоми з найбільшим значенням функції пристосованості, або здійснюється перехід до наступного кроку генетичного алгоритму, тобто до селекції. У класичному генетичному алгоритмі вся попередня популяція хромосом заміщається новою популяцією нащадків, що має ту ж чисельність.

Вибір «найкращої» хромосоми. Якщо умова зупинки алгоритму виконана, то слід вивести результат роботи, тобто представити шуканий розв'язок задачі. Кращим рішенням вважається хромосома з найбільшим значенням функції пристосованості.

4 ВИКОРИСТОВУВАНІ ТЕХНІЧНІ ЗАСОБИ

Мінімальна конфігурація ПЕВМ для роботи програми «Quiz» – процесор Intel Pentium 4 і вище з ОЗУ не менше 512МБ. На ПЕВМ повинен бути встановлений http-сервер Apache.

5 ВИКЛИК І ЗАВАНТАЖЕННЯ

Для виклику web-додатку необхідно вказати шлях до сторінки index.php у адресній строчці web-браузеру.

# 6 ВХІДНІ ДАНІ

Вхідними даними для роботи web-додатку є введені користувачем параметри для роботи генетичного алгоритму – точність розрахунку для кожного параметру, та кількість особин у популяції.

# 7 ВИХІДНІ ДАНІ

Вихідними даними є вивід на екран мінімумів обох функцій, а також значення параметрів, необхідних для отримання цих мінімумів.

# 8 Перелік використаних джерел інформації

1. Лесин В. В., Лисовец Ю. П. Основы методов оптимизации. / В. В. Лесин, Ю. П. Лисовец - М.: Изд-во МАИ, 1995. 344 с.
2. ПанченкоТ. В. Генетические алгоритмы: учебно-методическое пособие / под ред. Ю. Ю. Тарасевича. - Астрахань : Издательский дом «Астраханский университет», 2007. - 87 с.
3. Гладков Л.А., Курейчик В.В., Курейчик В.М. Генетические алгоритмы/ Под ред. В.М. Курейчика. - 2-е изд., испр. и доп. - М.: ФИЗМАТЛИТ, 2006. — 320 с.
4. Батищев, Д.И. Генетические алгоритмы решения экстремальных задач/ Д.И. Батищев ; Нижегородский госуниверситет. - Нижний Новгород : 1995. - 62с.
5. Рутковская, Д. Нейронные сети, генетические алгоритмы и нечеткие системы/ Д. Рутковская, М. Пилиньский, Л. Рутковский – М. : Горячая линия – Телеком, 2006. – 452 с.
6. Тим Джонс, М. Программирование искусственного интеллекта в приложениях/ М. Тим Джонс – М. : ДМК Пресс, 2004. – 312 с.