

Севідов Павло Миколайович

студент

Національний технічний університет України «Київський політехнічний інститут»

Севидов Павел Николаевич

студент

Национальный технический университет Украины «Киевский политехнический институт»

Sevidov P.

student

National Technical University of Ukraine «Kyiv Polytechnic Institute»

**ДОСЛІДЖЕННЯ АЛГОРИТМІВ ОПТИМІЗАЦІЇ ІНДИВІДУАЛЬНИХ
КАЛЕНДАРНИХ ПЛАНІВ НАВЧАЛЬНОГО ПРОЦЕСУ**

**ИССЛЕДОВАНИЕ АЛГОРИТМОВ ОПТИМИЗАЦИИ ИНДИВИДУАЛЬНЫХ
КАЛЕНДАРНЫХ ПЛАНОВ УЧЕБНОГО ПРОЦЕССА**

ALGORITHMS FOR INDIVIDUAL TRAINING SCHEDULE OPTIMIZATION

Анотація. У даній роботі ставиться завдання розглянути алгоритми оптимізації індивідуальних планів-графіків навчання, для вирішення проблеми управління часу студента. Було досліджено оптимальність використання генетичних алгоритмів для вирішення задачі багатокритеріальної оптимізації календарного плану навчального процесу.

Ключові слова: алгоритми оптимізації, генетичний алгоритм, план-графік, фітнес функція, алгоритм імітації вогпалу, жадібний алгоритм.

Аннотация. В данной работе ставится задача рассмотреть алгоритмы оптимизации индивидуальных план-графиков обучения, для решения проблемы управлением времени студента. Было исследовано оптимальность использования генетических алгоритмов для решения задачи многокритериальной оптимизации календарного плана учебного процесса.

Ключевые слова: алгоритмы оптимизации, генетический алгоритм, план-график, фитнес функция, алгоритм имитации отжига, жадный алгоритм.

Summary. In this paper was considered the task of optimization algorithms of individual training schedules to solve the problem of student time management. Investigated the optimal use of genetic algorithms for solving the problem of multi-criteria optimization schedule of the educational process.

Key Words: optimization algorithms, genetic algorithm, a schedule, a fitness function, simulated annealing, a greedy algorithm.

Люди протягом усього свого життя стикаються з задачею оптимального планування свого часу. Кожен з щодня займається плануванням власного часу. Майже всі люди керуються при цьому «схожим алгоритмами», намагаючись виконати всі важливі справи та вчасно, виділивши при цьому максимальний час на дозвілля та відпочинок.

Різноманітність в навчальному плані предметів і видів навантаження призводить до неоптимального використання студентом власного часу. Наслідком даної проблеми є так званий «синдром студента». Це призводить до втрати часу, яке виділяється студентом

при оцінці трудомісткості і ризиків навчальних робіт, і до збільшення рівня стресу в кінці семестру. Рішення даної проблеми є актуальним і дозволить зменшити навантаження на студентів і викладачів. Найвні рішення не враховують індивідуальний розклад студента і рівень його щоденної завантаженості власними справами при формуванні графіка виконання навчального плану.

Даний тип задач можна віднести до задач з теорії розкладів. Суть якого полягає в створенні певного порядку дій та оптимальному розподілі ресурсів (вільний час студента), що необхідні для досягнення

поставленої мети (успішне та своєчасне виконання студентом усіх робіт, що передбачені навчальним процесом з врахуванням власного графіку).

Критерії до задачі

Складання індивідуального розкладу для студента передбачає ряд критеріїв, з яких виділено два основних типи: *hard stop* (критерії, що заборонено порушувати для даної задачі) і *soft stop* (критерії, що не бажано порушувати, порушення призводить до погіршення результату).

До *Hard Stop* критеріїв віднесено:

1. Кожна робота повинна бути закінчена і здана до крайнього строку (*deadline*).
2. Виконану роботу можна здати за наявності запиття з викладачем, який приймає роботу.
3. Початок виконання роботи можливо за умови, якщо немає інших справ в цей час.

Перелік *Soft Stop* критеріїв:

1. Показник завантаженості не повинен перевищувати допустиму норму.
2. Задачі з високим пріоритетом, виконуються якомога раніше.
3. Рівномірність завантаженості кожного робочого дня.
4. Сума дат завершення робіт є мінімальною.

Вибір даного набору критеріїв є критичним для поставленої задачі. Наприклад, порушення крайнього строку суперечить меті – виконання всіх робіт вчасно, використовуючи мінімальну кількість ресурсів. Перелік *soft stop* критеріїв представлений в порядку ваги критичності порушення. Тобто порушення норми завантаженості, призводить роботи поза нормований робочий день і зменшенню продуктивності. Мінімальна сума дат завершення робіт – показник знаходження оптимального рішення. Результатом може бути знайдене гарне рішення, що не суперечить всім критеріям, крім останнього. Тобто порушення даного критерію не є критичним.

Пошук рішення поставленої задачі можливо шукати на базі таких існуючих методів: евристичних, стохастичних, детермінованих. Даний тип завдань є NP складною і не існує точної математичної моделі для знаходження оптимального рішення. Тому доцільно використання стохастичних і евристичних методів. Для дослідження розглядалися алгоритми: імітації відпалу, як представник стохастичного методу, генетичний алгоритм та жадібний алгоритм – евристичного.

Алгоритми оптимізації

Жадібний алгоритм. Простий і прямолінійний евристичний алгоритм, який приймає найкраще рішення, виходячи з наявних на поточному етапі даних,

не турбуючись про можливі наслідки, сподіваючись врешті-решт отримати оптимальне рішення. Легкий в реалізації і часто дуже ефективний за часом виконання. Мінус – багато задач не можуть бути розв’язані з його допомогою.

Алгоритм імітації відпалу. Алгоритм ґрунтується на імітації фізичного процесу, який відбувається при кристалізації речовини, в тому числі при відпалі металів. Передбачається, що атоми вже вишикувалися в кристалічну решітку, але ще допустимі переходи окремих атомів з одного осередку в іншу. Передбачається, що процес протікає при поступово знижується температурі. Перехід атома з одного осередку в іншу відбувається з певною ймовірністю, причому ймовірність зменшується з пониженням температури. Стійка кристалічна решітка відповідає мінімуму енергії атомів, тому атом або переходить в стан з меншим рівнем енергії, або залишається на місці.

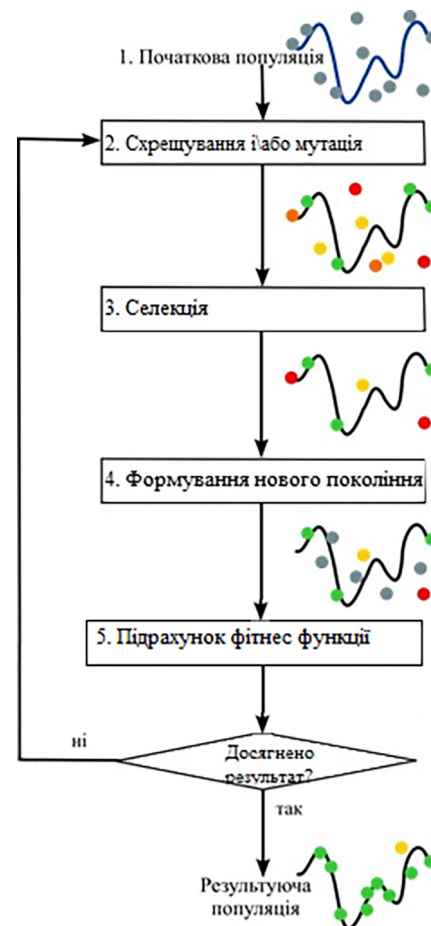


Рис. 1. Блок-схема роботи генетичного алгоритму

Генетичний алгоритм. Завдання формалізується таким чином, щоб її рішення могло бути закодовано у вигляді вектора («генотипу») генів, де кожен ген може бути бітом, числом або якимось іншим об’єктом. У класичних реалізаціях генетичного алгоритму (ГА) передбачається, що генотип має фіксовану

довжину. Однак існують варіації ГА, вільні від цього обмеження.

Деяким, звичайно випадковим, чином створюється безліч генотипів початкової популяції. Вони оцінюються з використанням «функції пристосованості», в результаті чого з кожним генотипом асоціюється певне значення («пристосованість»), яке визначає наскільки добре фенотип, їм описуваний, вирішує поставлене завдання.

Для поставленої проблеми, рішення за допомогою ГА критичним є вибір методів мутації і кросинговеру, також налаштування значень двох змінних ймовірності мутації і розміру популяції. Провівши дослідження було обрано:

- **Впорядкований кросинговер (OX1):** частина одного з батьків береться за основу нащадку. Далі в отриманий нащадок недостаючи гени беруться з другого батька зберігаючи порядок та пропускаючи ті, що вже є в частині від першого батька. Таким чином не порушуються жодне з обмежень.
- **Метод мутації – «обмін»**, що має найкращі показники простоти реалізації і швидкодії, суттю якого є зміна в послідовності двох сусідів одного випадково обраного гена.

Далі на основі тестових даних, для яких взято учбовий план зі всіма п'ятого семестру, зроблено моделювання індивідуальні план-графіки. Провівши аналіз індивідуальних розкладів з різними значеннями

параметрів розміру популяції і ймовірності мутацій, отримано такі результати.

Популяція складається з певної кількості різних особин. При збільшенні популяції, збільшується варіативність особин, що в свою чергу збільшує шанси досягти оптимального результату. Мінусом збільшення популяції є уповільнення роботи алгоритму, тому потрібно знайти компромісне рішення. Як бачимо з рисунку 2 таким значенням є розмір популяції в межах від 50 до 70.

При збільшенні ймовірності, збільшується «випадковість» генерації значень нового покоління, що в свою чергу унеможливає досягти оптимального значення. Результатом даних досліджень є вибір значення ймовірності мутації в межах від 0.1 до 0.25.

Результуючі результати значення функції оптимізації по відношенні до початкового приближення склали для досліджених алгоритмів:

1. Генетичний алгоритм – 26%.
2. Алгоритм імітації відпаду – 29%.
3. Жадібний алгоритм – 17%.

Висновки

Результатом даної роботи є аналіз особливостей та ефективності використання генетичного алгоритму для вирішення задачі створення та оптимізації календарного плану навчального процесу. Продуктом даного аналізу є порівняльна характеристика показників

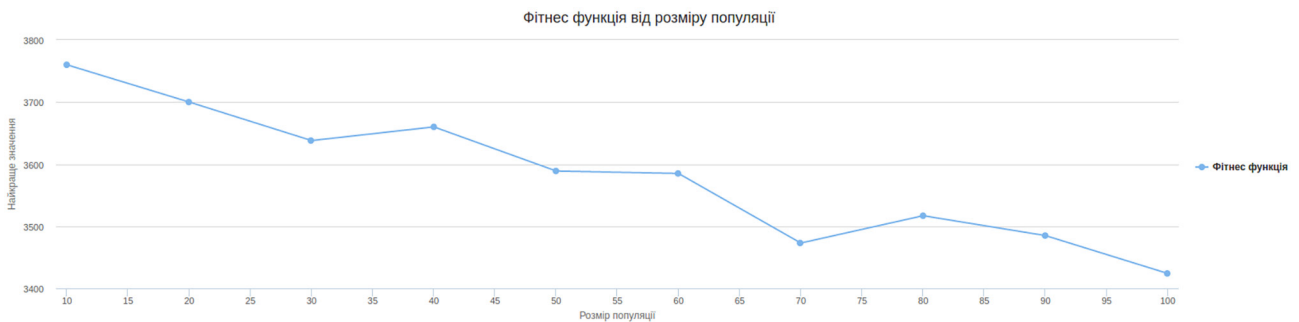


Рис. 2. Залежність найкращого значення фітнес функції від розміру популяції



Рис. 3. Залежність найкращого значення фітнес функції з популяції до ймовірності мутації



Рис. 4. Порівняння значень функцій оптимізації

роботи та результатів оптимізації розкладу на один навчальний семестр.

Варіації етапів генетичного алгоритму дали можливість підібрати варіант, що не порушує наявних критеріїв. Було обрано впорядкований кросингвер (OX1). В якому частина одного з батьків береться за основу нащадку. Далі в отриманий нащадок недостаючи гени беруться з другого батька зберігаючи порядок та пропускаючи ті, що вже є в частині від першого батька. Таким чином не порушуються жодне з наявних критеріїв. Найкращими показниками простоти реалізації і швидкодії відповідає метод мутації обміну, що змінює в послідовності двох сусідів одного випадково обраного гена.

Для генетичного алгоритму було відображено залежність параметрів від найкращого значення функції. При збільшенні розміру популяції, зростає ймовірність досягнути оптимального рішення і досягається краще значення фітнес функції. Генетичний алгоритм має явну перевагу, використовуючи його створення

індивідуального план-графіку на базі для веб-додатку тому, що отримання «гарного» рішення досягнуто приблизно на 200 ітерації роботи і витрачається мала кількість часу.

Недоліки алгоритму: час виконання функції оптимізації велике, необхідність знайти всі рішення задачі, а не одне з них, багато тимчасових даних, недоведеність збіжності.

Також було наведено порівняльну характеристику з алгоритмом імітації відпалу. Фінальна оптимізація генетичного алгоритму менша на 3%. Можливо досягнути кращу мінімізацію збільшуючи розмір популяції, але постає інша проблема — задіяння неймовірної кількості пам'яті. Це призводить до частих запусків збирача сміття, в результаті чого «просідання» в швидкості видачі готового результату кінцевому користувачеві.

Надалі можливо поліпшити результати, використовуючи гібридні алгоритми, комбінуючи переваги розглянутих алгоритмів.

Література

1. Лазарев А. А. Решение NP-трудной задачи теории расписаний минимизации суммарного запаздывания // Журнал Вычислительной математики и математической физики — 2007. том 47, N.6. — С. 1087–1098.
2. Лазарев А. А., Гафаров Е. Р. Теория расписаний. Минимизация суммарного запаздывания для одного прибора. // Научное издание, М.: Вычислительный центр им. А. А. Дороницына РАН, 2006. — 134 с.
3. Alon N., Woeginger G.J., Yadid T. Approximation schemes for scheduling on parallel machines // J. of Scheduling. — 1998. — V. 1. — P. 55–66.
4. Brucker P., Knust S. Complex scheduling Springer-Verlag Berlin, Heidelberg, Germany, 2006.
5. Nouredin Sadawi [Електронний ресурс]. — Режим доступу: <https://www.youtube.com/channel/UCNYv4HA3WjV3gZGLfBehRWQ>. — Дата доступу: 27.05.2016.