

Paweł Hoser¹, Luiza Ochnio²✉

Szkoła Główna Gospodarstwa Wiejskiego w Warszawie

Wykorzystanie systemu wieloagentowego do optymalizacji planu zajęć na uczelni

University schedule optimisation with the use of a multi-agent system

Synopsis. Jednym z problemów natury logistycznej, z którym zmagają się planiści jest takie zoptymalizowanie planu zajęć akademickich, by spełniał on wszystkie wymagane kryteria i jednocześnie był satysfakcjonujący dla użytkowników. Wielu autorów przy projektowaniu narzędzia do wspomagania wykonania tego zadania próbuje wykorzystywać różnego rodzaju algorytmy i metody z różnym powodzeniem. W pracy zaproponowano własną aplikację, której automatyzacja może być oparta na systemie wieloagentowym, która jest dostosowana do szczególnych warunków pracy na Wydziale Zastosowań Informatyki i Matematyki SGGW. Aplikację tę można będzie rozszerzyć do pracy na całej uczelni, jak i nawet na innych uczelniach i szkołach. Pomysł automatycznego wspomaganie powstał w trakcie praktycznego korzystania z aplikacji podczas układania rzeczywistych planów zajęć. Wyniki obecnych eksperymentów wydają się obiecujące.

Słowa kluczowe: plan zajęć, metody heurystyczne, optymalizacja, systemy wieloagentowe

Abstract. One of the logistical problems faced by planners is optimising the schedule of academic classes to meet all the required criteria and, at the same time, be satisfactory for users. When designing a tool to support this task, many authors try to use various types of algorithms and methods with varying degrees of success. This study proposes its own application whose automation can be based on a multi-agent system, which is adapted to the specific working conditions at the Faculty of Applied Informatics and Mathematics of the Warsaw University of Life Sciences. This application will be able to be extended to work at the entire university, as well as at other universities and schools. The idea of automatic support arose during the practical use of the application while arranging real schedules. The results of the current experiments seem quite promising.

Key words: Scheduling, heuristic methods, optimisation, multi-Gantt systems

Kody JEL: O14, C88

¹Paweł Hoser – Szkoła Główna Gospodarstwa Wiejskiego w Warszawie; Katedra Sztucznej Inteligencji; Instytut Informatyki Technicznej; e-mail: pawel_hoser@sggw.edu.pl; <https://orcid.org/0000-0003-4409-8989>

²✉ Luiza Ochnio – Szkoła Główna Gospodarstwa Wiejskiego w Warszawie; Katedra Logistyki; Instytut Ekonomii i Finansów; e-mail: luiza_ochnio@sggw.edu.pl; <https://orcid.org/0000-0001-8875-7945>

Wstęp

Planowanie zajęć uniwersyteckich jest złożonym problemem i nie można go rozwiązać, stosując tylko kilka ogólnych zasad czy kryteriów. Jak zauważa wielu autorów, nie ma jednego idealnego narzędzia i sposobu ułożenia planu, a poziom trudności i poszukiwanie rozwiązań w oparciu o podobne algorytmy zbliżony jest do optymalizacji planowania linii kolejowych [Wagner 2003, Goosuerw 2004, Liebchen 2008, Corman and Meng 2014]. W poszukiwaniu najlepszej metody do zautomatyzowania procesu układania planu na uczelniach, wielu autorów proponuje użycie różnych technik, narzędzi i algorytmów. Chan i inni [2002] opisał zautomatyzowany system planowania programu nauczania oparty na metodologii wyszukiwania stochastycznego, czyli algorytmie koewolucyjnym. Wyniki wykazały, iż system ten jest korzystniejszy w porównaniu z systemem ręcznym, jednak nie do końca zadowalający. Skomplikowane relacje między okresami, przedmiotami i salami wykładowymi utrudniają uzyskanie wykonalnego rozwiązania. Dlatego znalezienie dobrze funkcjonującego narzędzia dla danej uczelni jest ciągle trudnym problemem. W swoim artykule Irene i inni [2009] przedstawili hybrydowy algorytm optymalizacji roju cząstek w celu rozwiązania problemu planowania zajęć na uniwersytecie. Proponowane podejście (optymalizacja hybrydowego roju cząstek z rozumowaniem opartym na ograniczeniach) wykorzystuje optymalizację roju cząstek do znalezienia położenia pomieszczenia i szczeliny czasowej za pomocą odpowiedniej funkcji celu, a do wyszukania najlepszej wartości preferencji pojemności sali na każde zajęcia w rozsądnym czasie obliczeniowym zostało wykorzystane rozwiązanie oparte na ograniczeniach. Ten pomysł rozwiązania także ma swoje wady i wymagał dalszego udoskonalenia. Wiele autorów proponuje podejście heurystyczne w tym algorytmy genetyczne w celu zmniejszenia ilości konfliktów i optymalizacji dopasowania z uwzględnieniem liczby studentów na poszczególnych zajęciach, czasu zajęć, wielkości sal, dostępności czasowej w poszczególnych zajęciach wykładowcy prowadzącego zajęcia [Deris i in. 1999, 2000, Mittal i in. 2015, Ahmad i in. 2018, Prosad i in. 2022]. Nie wszystkim jednak udaje się objąć algorytmem wszystkie kryteria, które należy potem dodawać ręcznie. W Szkole Głównej Gospodarstwa Wiejskiego w Warszawie także powstał pomysł zautomatyzowania planowania. Jako że, w tym przypadku jesteśmy zarówno planistami (czyli użytkownikami), jak i programistami potrafiliśmy stworzyć bardzo wygodne dla siebie narzędzie komputerowe do układania planów. Narzędzie to szczególnie nadaje się do układania planów na Wydziale Zastosowań Informatyki i Matematyki SGGW. Każda jednostka ma bowiem swoją specyficzną charakterystykę. Sposób układania planów na jednym wydziale różni się od układania planu na innym wydziale, a tym bardziej na innej uczelni. Nie mniej jednak, jesteśmy przekonani, że aplikację można skutecznie rozszerzyć tak, się dobrze nadawała się do układania planów na innych wydziałach naszej uczelni. Już od siedmiu lat, plany zajęć na wspomnianym wydziale są układane w tejże właśnie aplikacji, co spowodowało, że mamy już pewne doświadczenie. Jest więc dość naturalne, że następnym krokiem powinny być próby stworzenia komputerowej automatyzacji układania planu. Celem pracy jest zaproponowanie innego podejścia opartego na systemie wieloagentowym do zautomatyzowania i zoptymalizowania układania planu zajęć.

Metodyka badań

W celu automatyzacji i optymalizacji planu zajęć na uczelni wyższej autorzy zwrócili uwagę na możliwość wykorzystania swoistego rodzaju algorytmów w celu zaimplementowania własnej aplikacji. Aplikacja służąca do wygodnego i efektywnego układania planu zajęć musiałaby spełniać wiele warunków, ograniczeń i powinna posiadać wbudowane mechanizmy unikania kolizji. Podczas poszukiwania inteligentnych metod automatycznego układania planu, z uwag na złożoność problemu, zdecydowano się na zastosowanie systemów wieloagentowych jako metody heurystycznej [Weiss 1999]. Pomysł ten narodził się właśnie podczas układania rzeczywistych planów zajęć. Wykorzystywane w pracy systemy wieloagentowe jako złożone i rozproszone należą do zbioru metod sztucznej inteligencji. W rozpatrywanym przypadku każdy agent byłby względnie uproszczony, logiczny. W procesie optymalizacji zaimplementowano metaheurystykę symulowanego wyżarzania [Rutkowski 2006].

Ograniczenia i preferencje planu zajęć

Wszyscy studenci biorący udział w zajęciach są podzieleni zgodnie z drzewem roczników. Najmniejszą jednostką jest grupa laboratoryjna, a w przypadku fakultetów, to jest grupa fakultetowa. Dwie grupy laboratoryjne składają się na jedną grupę ćwiczeniową. Grupy laboratoryjne są składane parami w strukturze drzewa roczników i w ten sposób powstają grupy ćwiczeniowe. Jeśli liczba grup laboratoryjnych na roczniku jest nieparzysta, to siłą rzeczy, jedna grupa ćwiczeniowa musi się składać tylko z jednej grupy laboratoryjnej. Każdy rocznik jest podzielony na grupy laboratoryjne, a w przypadku starszych roczników, grupy mają jeszcze swoje specjalizacje. Najbardziej fundamentalne ograniczenia w planie, to kolizje sal, kolizje prowadzących i kolizje grup. Takie kolizje nie mogą występować w poprawnie ułożonym planie zajęć. W każdej sali mogą się odbywać tylko jedno zajęcia w tym samym czasie. Każdy prowadzący może prowadzić tylko jedno zajęcia w tym samym czasie. Każda grupa studencka może mieć tylko jedno zajęcia w tym samym czasie.

Dwa zajęcia $z_i = (g_i, pd_i, tzi, pr_i, s_i, t_i, d_i)$ i $z_j = (g_j, pd_j, tz_j, pr_j, s_j, t_j, d_j)$ pokrywają się czasowo, jeśli ich interwały czasowe mają część wspólną, czyli:

$$[t_i, t_i + d_i] \cap [t_j, t_j + d_j] \neq \emptyset \quad (1)$$

co jest tu równoznaczne ze stwierdzeniem, że: $(t_j < t_i + d_i) \wedge (t_j + d_j > t_i)$.

Wówczas, dla $pr_i = pr_j$ kolizja prowadzącego, dla $s_i = s_j$ kolizja sali, a dla $g_i = g_j$ grupy.

Przy czym, przedziały czasu są tu matematycznie traktowane jako przedziały otwarte, co tutaj wiąże się z tym, że może nie być żadnej przerwy między zajęciami. W przypadku nakładania się czasowego zajęć, może dochodzić do kolizji prowadzącego, sali lub grupy.

$$\begin{cases} (t_j, < t_i + d_i) \wedge (t_j + d_j > t_i) \wedge (pr_i = pr_j) \Rightarrow KP \\ (t_j, < t_i + d_i) \wedge (t_j + d_j > t_i) \wedge (s_i = s_j) \Rightarrow KS \\ (t_j, < t_i + d_i) \wedge (t_j + d_j > t_i) \wedge (g_i = g_j) \Rightarrow KG \end{cases} \quad (2)$$

Należy jeszcze zaznaczyć, że kolizje grup mogą dotyczyć kolizji różnych typów zajęć (wykład, ćwiczenia, laboratorium, seminarium, fakultet) i w takich kolizjach może brać udział większa liczba grup. Ważne jest, że każda grupa studentów nie może mieć dwóch różnych zajęć w tym samym czasie. Oczywiście fundamentalnym wymogiem jest także to, żeby wszystkie zajęcia zadeklarowane do ułożenia znalazły się w planie.

Dodatkowo plan zajęć musi spełniać wiele innych wymogów. Zajęcia laboratoryjne muszą się odbywać oddzielnie dla każdej grupy laboratoryjnej. Wyjątkiem jest sytuacja, w której liczba wszystkich studentów nie przekracza 20 osób. Zajęcia ćwiczeniowe muszą się odbywać oddzielnie dla każdej grupy ćwiczeniowej, czyli dla dwóch grup laboratoryjnych. Wyjątkiem jest sytuacja, w której łączna liczba studentów nie przekracza 40 osób. Wykłady dla przedmiotów podstawowych muszą się odbywać łącznie dla całego rocznika w tym samym czasie w odpowiednio dużej sali. Wykłady przedmiotów kierunkowych natomiast muszą odbywać się łącznie dla wszystkich specjalizacji, których dotyczy ten przedmiot. Terminy zajęć muszą być takie, żeby nie pokrywały się z ważnymi zajęciami osób prowadzących, takimi jak spotkania rady wydziału, spotkania komisji senatu, seminaria katedry lub instytutu, a także ustalone zajęcia na innych wydziałach naszej uczelni. Oprócz tego, wszystkie zajęcia mogą się tylko odbywać od godziny 8:00 do godziny 20:00, każdego dnia w tygodniu.

Kolizje, które są ograniczeniami ułożenia planu, możemy podzielić na dwie kategorie. Kolizje twarde, które są absolutnie wykluczone i kolizje miękkie, które ostatecznie mogą być chwilowo zaakceptowane, ale jest to bardzo niekorzystne. Kolizje twarde, to kolizje sal, prowadzących i grup, a kolizje miękkie, to kolizje z zajęciami innych wydziałów i kolizje z ważnymi spotkaniami osób prowadzących. Oprócz tego, bardzo ważną rzeczą są preferencje studentów i osób prowadzących, ponieważ bardzo się staramy, aby plan ten był jak najlepszy dla wszystkich. To głównie pod tym kątem wykonywana jest optymalizacja ułożenia planu.

Preferencje studentów głównie polegają na minimalizacji tak zwanych okienek czasowych. Zakłada się, że przerwy między zajęciami powinny trwać 15 minut, to jest sytuacja optymalna, wyjątkiem są sytuacje, kiedy studenci i prowadzący muszą przechodzić do innego budynku. Im dłuższa przerwa między zajęciami (w ciągu jednego dnia), tym gorzej. Drugą istotną sprawą jest liczba zajęć w ciągu dnia. Zakłada się, że optymalną sytuacją dla studentów jest od 6 do 8 godzin zajęć w ciągu jednego dnia. Akceptowalną sytuacją są też jeszcze 4 godzinny dziennie, ale nie mniej. Studenci nie powinni przyjeżdżać na tylko jedno zajęcia jednego dnia. Podobnie, 10 godzin zajęć jednego dnia jest jeszcze sytuacją dopuszczalną, ale więcej niż 10 już nie. W przypadku tzw. okienek dopuszczalna jest jedna dłuższa przerwa w środku dnia między zajęciami, jeżeli zajęć w danym dniu jest dużo, jest to planowana przerwa na obiad i odpoczynek. W przypadku osób prowadzących, podobnie zakłada się, że optymalne przerwy między zajęciami powinny być 15 minutowe i korzystna może być też dłuższa przerwa obiadowa. Jednak optymalna liczba godzin w ciągu dnia jest mniejsza dla prowadzącego niż dla studentów. Za to dopuszcza się dłuższe okienka dla osób prowadzących. Oprócz tego osoby prowadzące mogą zgłosić swoje preferencje, w naszym przypadku, polegają one na zgłoszeniu jednego dnia w tygodniu (lub dwóch dni, gdy zajęć jest mało) kiedy nie chcą prowadzić zajęć. Wszystkie te dane dotyczące preferencji są brane pod uwagę w procesie optymalizacji planu zajęć na studiach dziennych, na studiach zaocznych plan musi być dosto-

sowane do ilości zjazdów. Planiści przy układaniu planu ręcznie starają się jak najlepiej spełnić wszystkie oczekiwania studentów i prowadzących, z kolei w przypadku automatycznej optymalizacji planu, potrzebna jest ściśle zdefiniowana funkcja celu.

Algorytm optymalizacji planu zajęć

Zadaniem algorytmu optymalizacji planu zajęć jest wspomaganie pracy planisty. W proponowanej metodzie zakłada się, że algorytm wspomagający zaczyna działać w chwili, gdy plan zajęć jest już ułożony bezkolizyjnie. Zadaniem algorytmu jest poprawienie jakości planu względem tego, co ułożył już ręcznie planista. Algorytm optymalizacji jest oparty na idei systemu wieloagentowego. System wieloagentowy przetwarza dane w pewnym ustalonym środowisku i składa się z wielu agentów. Działanie całego systemu polega na działaniach wszystkich agentów [Huhns i Stephens 1999]. Najczęściej agenci są programami komputerowymi lub podprogramami większego programu i posiadają swój stan wewnętrzny definiowany przez ustalony układ parametrów. Działania agentów polegają, przede wszystkim, na zmienianiu środowiska, a także na zmianach swojego własnego stanu, jak i na wysyłaniu informacji do innych agentów [Papadimitriou i Płoski 2012]. Agenci podejmują swoje akcje na podstawie analizy wybranego obszaru środowiska, swojego stanu i informacji od innych agentów. Ogólnie zakłada się, że agenci powinni być jak najbardziej niezależni i autonomiczni [Rutkowski 2012]. Zakłada się też, że żaden poszczególny agent nie ma dostępu do całego systemu. System wieloagentowy powinien być maksymalnie zdecentralizowany i rozproszony [Flasiński 2011]. W przypadku systemu wspomaganie układania planu zajęć agenci są przeznaczeni do wykonywania zmian w planie i reprezentują wybrane grupy ludzi korzystających z planu. Są to agenci reprezentujący roczniki, grupy i osoby prowadzące, a także agent sali i agent główny. Każdy agent ma swoją własną funkcję celu, która reprezentuje interes osób z nim związanych. Cały system działa w sposób cykliczny podczas procesu optymalizacji, w trakcie procesu system wykonuje wiele cykli. Działania agentów polegają, przede wszystkim, na przenoszeniu zajęć w planie, co też wiąże się nieraz z wyborem innej sali i czasem też z wyborem innej osoby prowadzącej. Elementarne działanie agenta, polegające na przesunięciu jednego zajęcia, nazywamy jednym ruchem agenta. Działania agentów dzielimy na różne fazy w trakcie każdego cyklu – jest to faza gorąca, faza chłodzenia, faza zimna oraz porządkowanie sal i próby stosowania gotowych schematów. Cały proces optymalizacji odbywa się cyklicznie, aż do spełnienia warunku stopu.

Każda grupa laboratoryjna ma swojego agenta, który próbuje tylko optymalizować zajęcia laboratoryjne tej jednej grupy. Każdy rocznik ma też swojego agenta, ten agent próbuje optymalizować zajęcia całego rocznika, głównie zajmując się wykładami, które dotyczą wszystkich grup studenckich na jednym roczniku. Agent rocznika ma jeszcze za zadanie optymalizować wykłady przedmiotów kierunkowych, na które uczęszczają tylko studenci wybranych specjalizacji, a więc tylko niektóre grupy studentów. Dodatkowo jeszcze agent rocznika zajmuje się też zajęciami ćwiczeniowymi, gdzie grupy studenckie są łączone parami. Na Wydziale Zastosowań Informatyki i Matematyki SGGW zajęć ćwiczeniowych jest bardzo mało, głównie są one na pierwszym roku studiów. Zdecydowaną większością są zajęcia wykładowe i laboratoryjne. Agenci roczników podejmują jeszcze inne pewne specyficzne akcje pod koniec każdego cyklu. Agenci roczników próbują zastosować gotowe schematy i dotyczy to tylko związanych z nimi roczników. W przypadku rocznika mającego tylko jedną grupę laboratoryjną studenci są

reprezentowani tylko przez agenta grupy, bo w tej sytuacji agent rocznika jest już niepotrzebny. W przypadku, gdy przedmiot kierunkowy dotyczy tylko jednej grupy laboratoryjnej, wykładem zajmuje się agent grupy, a nie agent rocznika.

Lokalna i globalna funkcja celu

W całym procesie optymalizacji zakłada się, że szukamy takiego ułożenia planu, aby funkcja celu była jak największa, co odzwierciedla jak najlepsze zadowolenie wszystkich osób korzystających z planu zajęć. Argumentem funkcji celu jest jakieś ułożenie planu, a wartością funkcji celu jest liczba rzeczywista.

$$f: U \rightarrow R; y = F(x) \quad (3)$$

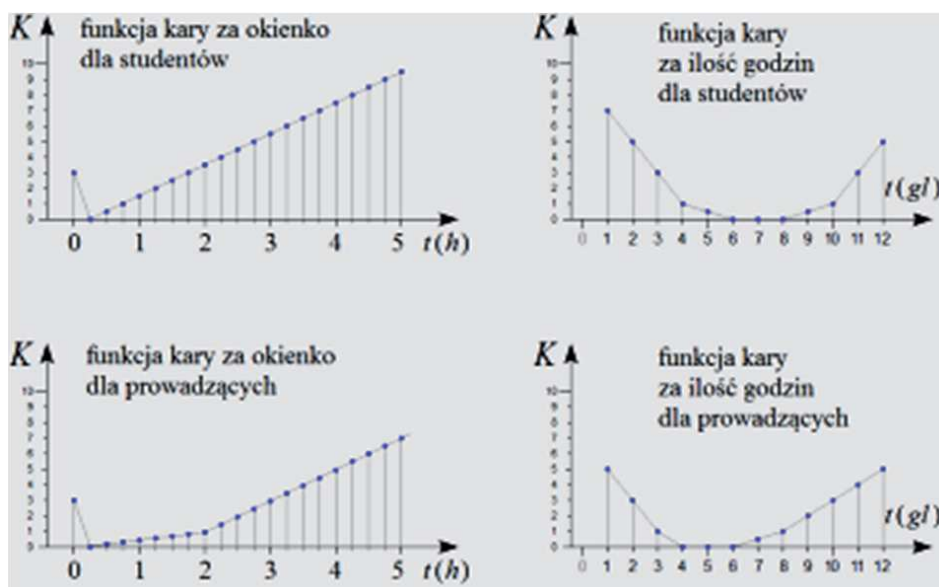
Gdzie U jest tutaj całą przestrzenią wszystkich bezkolizyjnych planów zajęć. W tej sytuacji, y jest liczbą rzeczywistą i jednocześnie wartością funkcji, a argument x jest jakimś konkretnym planem zajęć, czyli matematycznie wektorem parametrów wszystkich zajęć. Naszym zadaniem jest znalezienie takiego x , żeby wartość funkcji celu y była jak największa. W przypadku metod heurystycznych, z założenia, nie mamy żadnej gwarancji, że znajdziemy rzeczywiste maksimum w całej przestrzeni U , sukcesem jest znalezienie względnie dużego y .

W tym przypadku, wygodnie jest założyć, że wartość funkcji celu jest ujemną wartością funkcji kary f . Zatem optymalizacja polega na minimalizowaniu funkcji kary. Przyjmujemy, że plan idealny xI ma zerową wartość funkcji kary $f(xI) = 0$. Plan idealny to taki, gdzie nie występują żadne okienka ani dla studentów, ani dla prowadzących, liczba godzin zajęć dla studentów wynosi od 6 do 8 każdego dnia, a dla prowadzących od 4 do 6, oraz zajęcia są zgodne z preferencjami osób prowadzących, czyli prowadzący nie mają zajęć w niechciane dni. Oprócz tego w planie nie występują żadne kolizje (twarde i miękkie). Planów idealnych może być bardzo wiele. Każde odchylenie od tej sytuacji już idealne nie jest i wartość funkcji kary powinna być wtedy większa od zera. Im gorszy plan, tym wartość funkcji kary powinna być większa.

Warto tu jeszcze zaznaczyć, że taki plan idealny może być w ogóle nieosiągalny. Może być nawet tak, że żadnego planu bezkolizyjnego ułożyć się nie da. Wszystko zależy od danych wejściowych. Wystarczy bowiem, że przy dużej liczbie zajęć odpowiednio zmniejszymy liczbę sal lub liczbę prowadzących albo skrócimy całkowity czas i wtedy ułożenie planu będzie niewykonalne. Złożoność tego zagadnienia jest tak wielka, że możemy nawet nie wiedzieć o tym, czy ułożenie planu idealnego jest możliwe.

Celem jest, aby plan był jak najlepszy dla każdej grupy studentów i dla każdej osoby prowadzącej. Najwygodniej więc najpierw zdefiniować funkcje lokalne kary dla każdej grupy i dla każdego prowadzącego, a następnie je po prostu posumować. Początkowo trzeba ustalić, jak wartość funkcji kary ma zależeć od długości jednego okienka. Wartość funkcji kary dla grupy studentów będzie sumą kar za wszystkie okienka w ciągu tygodnia. Podobnie należy zrobić w przypadku wartości kary dla osoby prowadzącej. Jeszcze prościej jest w przypadku oceny liczby godzin w ciągu dnia. Trzeba ustalić wartości funkcji kary względem liczby godzin w jednym dniu. Wtedy wartość kary dla grupy lub dla prowadzącego jest sumą kar ze wszystkich dni tygodnia. Wartości funkcji kary są liczone w punktach. Wartości kar za okienka i liczby godzin zostały zdefiniowane funkcjami kawałkami

liniowymi. Rozważano użycie funkcji kwadratowych, ponieważ teoretycznie dla funkcji liniowych mogą powstawać rozwiązania bardzo niesprawiedliwe. Równie dobre jest jedno wielkie okienko dla jednej grupy i małe dla innych, jak rozwiązanie zrównoważone. Jednak właśnie równoległe działania agentów rozwiązują ten problem. W przypadku, gdy powstaje bardzo duże okienko dla jednej grupy, to agent tej grupy przerzuca część zajęć na inny dzień. Podobnie jest w przypadku agentów osób prowadzących. Na wykresach (rys. 1) przedstawiono zdefiniowane wartości funkcji kary.



Rysunek 1. Wykresy wartości funkcji kary dla okienek oraz ilości godzin na dzień (grupy i prowadzący)

Figure 1. Graphs of penalty function values for breaks and the number of hours per day (groups and lecturers)

Źródło: opracowanie własne.

Funkcje kary są trochę inne dla grup studenckich i dla osób prowadzących. Zakłada się, że osoba prowadząca powinna mieć optymalnie mniej zajęć dziennie niż student. Zakłada się też, że osoba prowadząca może bardziej tolerować okienka, które można traktować jako czas na konsultacje oraz przerwę na odpoczynek i obiad. W przypadku dodatkowych preferencji osób prowadzących trzeba zdefiniować funkcję kary dla zajęć zaplanowanych w dzień tygodnia niechciany przez prowadzącego. Ustalono, że za ułożenie takich zajęć naliczana jest kara 15 punktów – niezależnie już od ilości godzin.

Na koniec jeszcze należy uwzględnić problem zmiany sali dla osoby prowadzącej. Zakłada się, że prowadzący nie powinni zmieniać sali w ciągu dnia (zwłaszcza w krótkim czasie), jeśli nie zachodzi taka konieczność. Zmiana sali wiąże się oddawaniem i pobieraniem kluczy oraz przenoszeniem wszystkich potrzebnych rzeczy do prowadzenia zajęć przez prowadzącego. Tak więc, osoby prowadzące nie powinny zmieniać sali – tylko studenci. Odwrotnie było podczas pandemii, kiedy to zmiany sal przez studentów były traktowane jako niekorzystne i to było wtedy priorytetowe. Ustalono, że jeśli osoba prowadząca zmienia salę w ciągu krótkiej przerwy (do pół godziny) to naliczana jest kara jednego punktu do wartości kary osoby prowadzącej.

W ten sposób zostały zdefiniowane funkcje kary dla każdej grupy i dla każdego prowadzącego. Każda grupa jest związana z ustalonym podzbiorem zajęć i każdy prowadzący jest tak samo powiązany z ustalonym podzbiorem zajęć. Tak więc, zdefiniowane są wartości kary, dla i -tej grupy $K(G_i)$ oraz wartości kary dla j -tego prowadzącego $K(P_j)$. Są to wartości lokalnych funkcji kary.

Globalna funkcja kary jest sumą wartości kar dla wszystkich grup i prowadzących.

$$f(x) = \sum_{i=1}^{Ng} K(G_i) + \sum_{j=1}^{Np} K(P_j) \quad (4)$$

Globalna funkcja celu jest wartością ujemną funkcji kary $F(x) = -f(x)$.

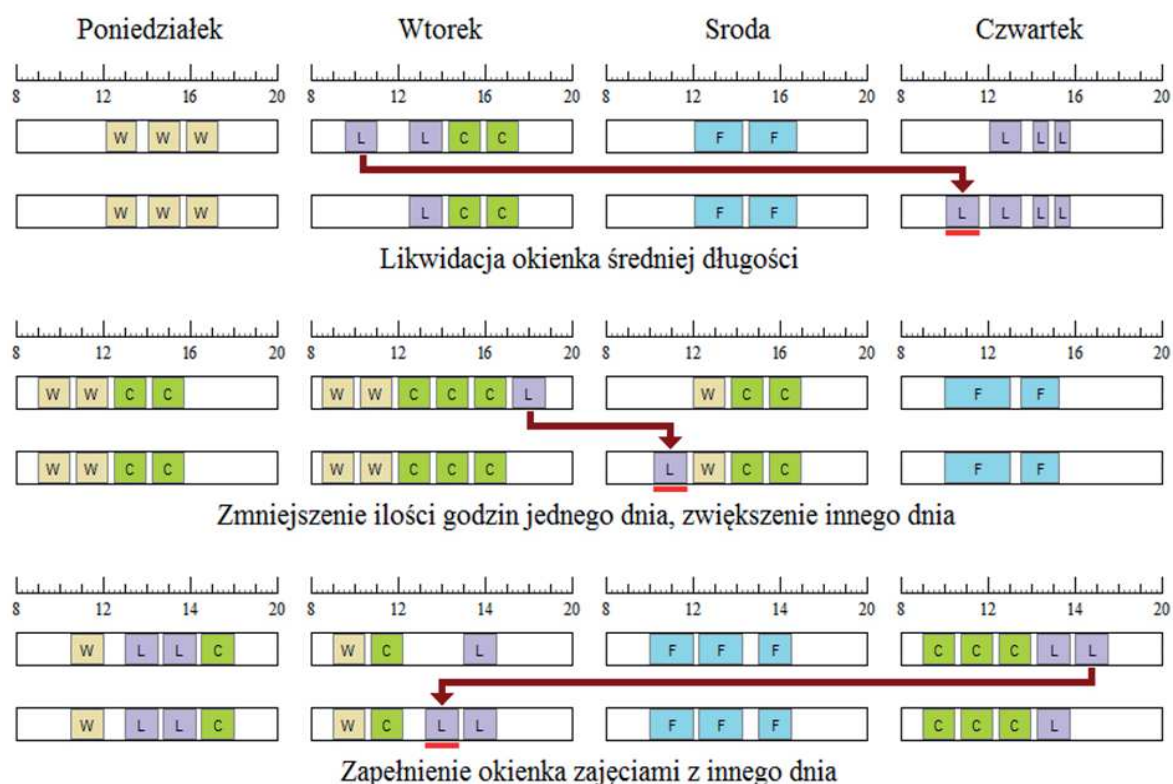
Działania agentów

Z założenia, system wieloagentowy powinien być systemem rozproszonym [Weiss 1999]. Jednak na razie program został napisany sekwencyjnie, więc działania agentów odbywają się po kolei. Harmonogram pojedynczych ruchów agentów musi być jakoś uporządkowany albo działania agentów są uruchamiane losowo. Istotne jest, że działanie systemu równoległego można symulować systemem sekwencyjnym, tylko działanie takie jest znacznie wolniejsze. Obecnie, działania agentów są uruchamiane po kolei w zadanym porządku. Jednak po każdym ruchu jednego agenta, następne posunięcie wykonuje już inny agent, co daje dobrą symulację równoległości. Jeśli w przyszłości program będzie zaprogramowany wielowątkowo, to będzie można naprawdę powiedzieć, że jest to system rozproszony i równoległy. Mimo to, ten system będzie wymagał pewnej koordynacji działań agentów i będzie to należeć do głównego wątku programu. Podczas każdego kroku, każdy agent wykonuje tylko jeden ruch i jest nim przesunięcie zajęć w planie. Tylko w przypadku stosowania gotowych schematów, działanie agenta rocznika będzie się składać z kilku przesunięć różnych zajęć. Podstawowe działania agentów grup, prowadzących i roczników są podzielone w całym cyklu na trzy fazy.

Podstawowe działanie agenta polega na przenoszeniu zajęć, czyli na zmianie dnia i godziny zajęć. Najpierw agent oblicza swoją funkcję kary dla zajęć dotyczących grupy osób, którą reprezentuje. Jeśli wartość kary jest mała, to agent nic nie robi w tym kroku, a więc jego ruch jest pusty. Jeżeli kara jest duża, to agent sprawdza przyczyny wysokiej wartości kary w obrębie swoich zajęć. To nie jest akurat bardzo trudne, wystarczy znaleźć większe okienka i dni, w których liczba godzin jest za mała lub za duża, a w przypadku agenta osoby prowadzącej, także należy znaleźć zajęcia, które są zaplanowane w dni lub pory dnia niechciane przez tego prowadzącego.

Po znalezieniu takiego miejsca w planie, agent próbuje naprawić ten problem poprzez przesunięcie jednego zajęcia na inny dzień lub na inną porę. Ogólnie robi to tak, żeby zlikwidować powód większej wartości funkcji kary. Tu jest wiele różnych możliwych przypadków, ale w każdym z nich łatwo napisać algorytm działania. Do tego typu działań, wystarczy nam nawet agent logiczny. Zakłada się, że po przesunięciu zajęcia, powinno ono być jak najlepiej dołączone do innych zajęć tej grupy, najlepiej z 15 minutową przerwą. Oczywiście, nie może to też powodować żadnej kolizji, czyli musi być dostępna sala

i prowadzący nie może mieć w tym czasie innych zajęć. Nieco inaczej jest w przypadku agenta prowadzącego, gdy zajęcie jest niezgodne z preferencjami, wtedy dla agenta prowadzącego, to ma priorytetowe znaczenie i też takiego ruchu nie wykona.



Rysunek 2. Przykłady działań agenta grupy w trzech różnych sytuacjach

Figure 2. Examples of group agent actions in three different situations

Źródło: opracowanie własne.

W przypadku agenta osoby prowadzącej akcje dotyczą wszystkich roczników, ale za to tylko tych zajęć, które ten prowadzący prowadzi. Pokazane na rysunku 2 przypadki działania agentów to tylko wybrane przykłady. Ogólnie, takich przypadków jest więcej, głównie różnią się od siebie konfiguracjami ułożonych zajęć dla jednej grupy lub dla jednego prowadzącego. Na przykład, przekładane zajęcie obok okienka może być ostatnie w ciągu dnia, a nie pierwsze (jak w tym przypadku) i tak dalej. Analogicznie, agent całego rocznika przekłada zajęcia, które są wspólne dla wielu grup, czyli wykłady, ćwiczenia, seminaria i fakultety (najczęściej jednak, są to wykłady). Efekt działania zilustrowano na planie zajęć (rys. 3).

W przypadku pierwszych roczników studiów pierwszego stopnia takich zajęć jest sporo, bo dochodzą do tego jeszcze wszystkie zajęcia ćwiczeniowe. Jednak to, że agent ma więcej zajęć do przeniesienia nie jest wcale dużym problemem. Zakładamy, że każda zmiana robiona przez agenta jest tak samo ważna, więc wszystko jedno którego to dotyczy przedmiotu. Obecnie zakładamy, że każdy agent jest tak samo ważny w całym systemie.

Dodatkowo w procesie optymalizacji trzeba uwzględnić problem optymalizacji wykorzystania sal. Do porządkowania sal został zdefiniowany „agent sal” w systemie. Agent ten pod koniec każdego cyklu stara się uporządkować zajętość sal. Najpierw sprawdza



Przeniesienie wykładu dla tego rocznika zlikwidowało luki obciążenia za jeden tydzień

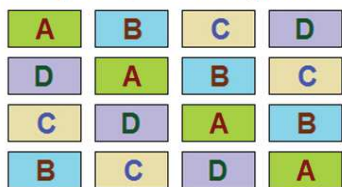
Rysunek 3. Przykład działania agenta rocznika – przeniesienie wykładu pozwoliło uniknąć okienek
 Figure 3. An example of agent actions of agent for the students year of study – moving the lecture avoid unnecessary breaks

Źródło: opracowanie własne.

występowanie niekorzystnych sytuacji, a potem próbuje je naprawić. Jeśli agent sal znajdzie sytuację, że prowadzący zmienia sale w krótkim czasie, to próbuje to naprawić poprzez zamianę sali z innym prowadzącym. W tym celu poszukuje innego prowadzącego, którego zajęcia można przenieść. Przy czym, zakłada się, że te inne zajęcia, które mają podlegać zamianie, muszą być jedynymi zajęciami tego drugiego prowadzącego w ciągu dnia. Oczywiście, warunkiem koniecznym jest ten sam typ obydwu sal, które mają być zamieniane. Inne przypadki nie są rozważane, bo generowało by to zbyt złożone problemy na tym etapie działania. Oprócz tego, staramy się również, żeby studenci nie musieli zmieniać sali w ciągu dnia bez żadnego powodu i to jest częściowo uwzględnione podczas każdego przenoszenia zajęć przez agenta grupy lub przez agenta rocznika.

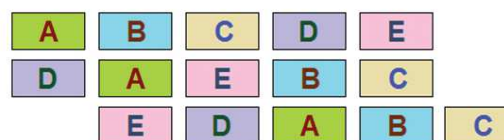
Ostatnim działaniem agentów są próby zastosowania gotowych schematów. Wykonują to agenci roczników, już na samym końcu każdego cyklu po uporządkowaniu sal przez agenta sal. Gotowe schematy, to gotowe kawałki bardzo dobrze ułożonego planu zajęć, zwykle dotyczą zajęć jednego rocznika w jednym dniu tygodnia. Cechą takich gotowych schematów ma być zerowa wartość funkcji kary w tej części planu.

cztery grupy , cztery przedmioty
 cztery sale , czterech prowadzących



plan idealny, brak okienek
 Zerowa wartość kary

trzy grupy , pięć przedmiotów
 trzy sale , pięciu prowadzących



studenci nie mają okienek, mają po 10 godzin
 prowadzący mają po jednym okienku
 i mogą mieć zajęcia na innych rocznikach

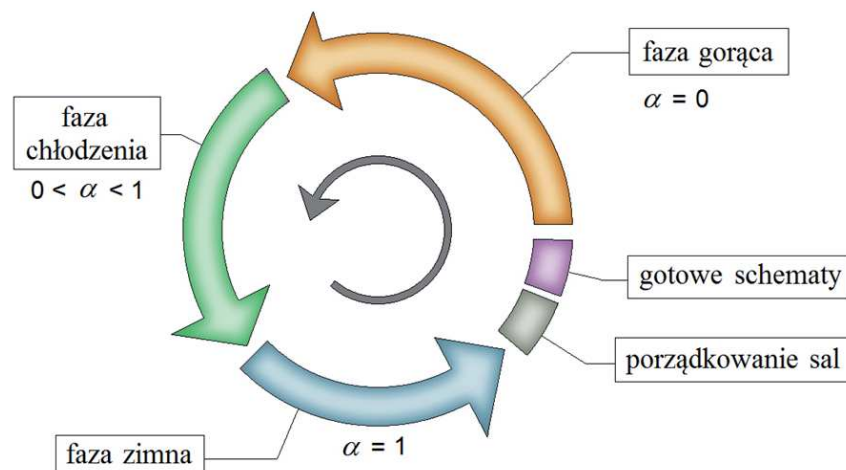
Rysunek 4. Dwa przykłady zastosowania gotowych schematów przez agenta rocznika
 Figure 4. Two examples of the use of ready-made schemes by the agent of the year of study

Źródło: opracowanie własne.

Na rysunku 4 przedstawiono dwa przykłady takich gotowych schematów, w tym przypadku dotyczą one laboratoriów, które są planowane dla każdej grupy oddzielnie. Podobnych schematów jest znacznie więcej, dotyczą one także zajęć ćwiczeniowych, wykładów lub seminariów i fakultetów. Warunkiem zastosowania takiego schematu jest odpowiednia dostępność zasobów w planie, a więc wolnych sal, osób prowadzących i wspólnego czasu dla wszystkich grup studenckich biorącym w tym udział. Bardzo istotne jest to, że stosowanie gotowych schematów jest wykonywane w sposób elastyczny. Nie jest wymagane, aby dostępność zasobów była całkowita, jeżeli niektóre zajęcia nie mogą być ułożone w schemacie z powodu kolizji, to schemat może być niepełny. Dopuszcza się, że 10% zajęć może brakować w takiej konfiguracji, nawet kosztem okienek w planie. Stosując permutacje zajęć w schemacie, można często tak zrobić, aby brakujące zajęcia znalazły się na zewnątrz takiego bloku zajęć i tak można pozbyć się okienek. Schematów może być bardzo dużo, można je nawet odpowiednio generować albo zapamiętywać najlepiej ułożone części planu, takie dla których wartość kary jest zerowa. To nasuwa jeszcze inny kierunek rozwoju inteligentnego wspomaganie układania planu. System, podczas każdego procesu układania planu, może stale zapamiętywać najlepsze fragmenty planu zajęć, żeby potem je stosować podczas dalszego działania. Co więcej, po takich udanych próbach, system może też zapamiętywać sytuacje, w których to miało miejsce. Następnie, system będzie rozpoznawał takie i podobne sytuacje w planie i będzie stosował gotowe sprawdzone schematy. Tak więc, w tym obszarze działania można spróbować zaimplementować system uczący się, oparty na różnych metodach sztucznej inteligencji.

Różne fazy działania agentów w jednym cyklu

Wszyscy agenci działają cyklicznie w kilku fazach po kolei. Jest to koordynowane przez głównego agenta całego planu. W fazie gorącej agenci podejmują działania, w ogóle nie biorąc pod uwagę funkcji celu innych agentów, czyli są tylko nastawieni na własny cel. Oczywiście powoduje to, że działanie agenta zmniejsza wartość jego funkcji kary, ale za to może zwiększać wartość kary innego agenta. Na przykład, każde zajęcie grupy ma jednocześnie swojego prowadzącego, przesunięcie zajęcia na korzyść grupy może być niekorzystne dla prowadzącego. Może to, na przykład, polegać na zwiększeniu jego okienka, bardziej niekorzystnym rozkładem godzin w ciągu tygodnia lub ustawieniem zajęcia w niechcianym czasie, czyli niezgodnie z preferencjami. Tak więc, w fazie gorącej każdy agent działa skrajnie egoistycznie. Tym samym agenci mają dużą większą swobodę ruchu, co przekłada się na więcej wykonanych akcji. Odwrotnie jest w fazie zimnej. W fazie zimnej agenci wykonują działania tylko pod warunkiem, że nie zwiększą wartości kary innym agentom. Przy czym, dwa punkty kary są tu wartością dopuszczalną, więc faza zimna nie jest też naprawdę idealnie zimną. W tym okresie, możliwości zmian są bardzo ograniczone. Faza chłodzenia jest formą przejściową, z fazy gorącej do zimnej. W trakcie działania agentów w fazie chłodzenia, wartość kary innych agentów jest brana pod uwagę losowo. Wprowadzony jest parametr $\alpha \in [0,1]$ który decyduje o prawdopodobieństwie brania pod uwagę wartości kar innych agentów. Podczas fazy chłodzenia parametr sterujący α jest stopniowo zmieniany od wartości 0 do 1. W ten sposób przechodzimy płynnie od fazy gorącej do zimnej. Można więc stwierdzić, że po prostu, podczas fazy gorącej $\alpha = 0$, a podczas fazy zimnej $\alpha = 1$.



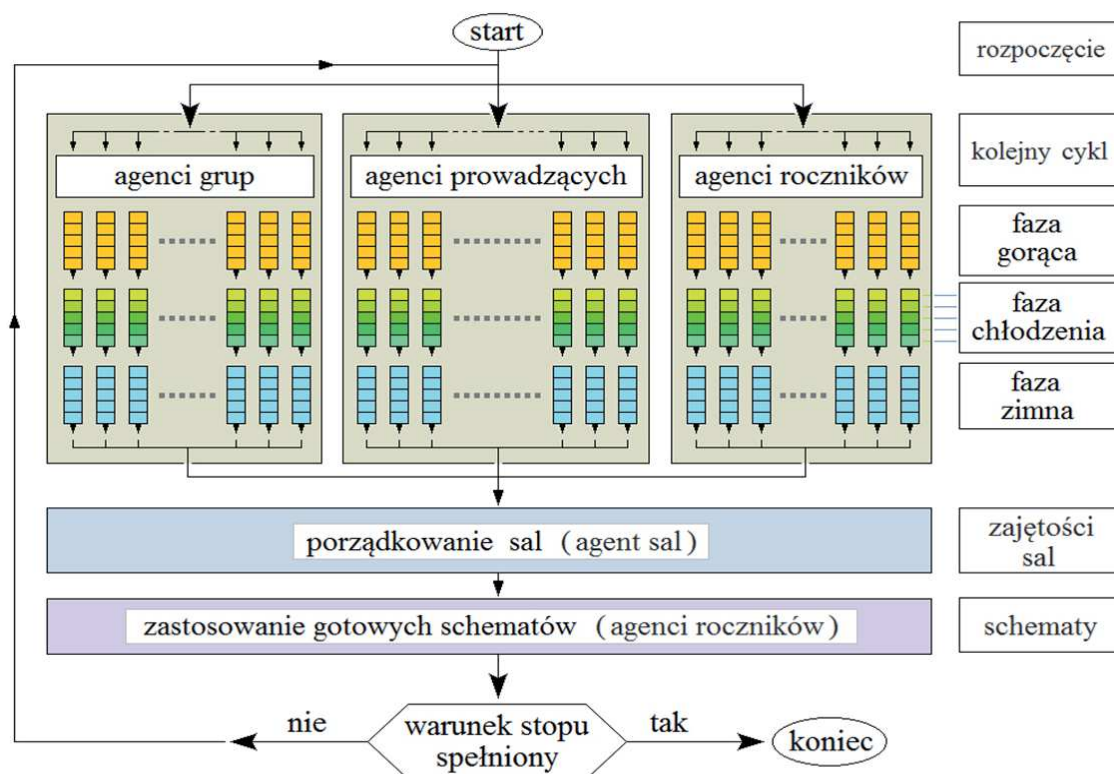
Rysunek 5. Cykl działania wszystkich agentów podczas każdego kroku procesu optymalizacji planu
Figure 5. The lifecycle of all agents activities during each step of the schedule optimization process
Źródło: opracowanie własne.

W ten właśnie sposób zaimplementowana jest metaheurystyka symulowanego wyżarzania, a cykl działania agentów podczas procesu optymalizacji przedstawiono na rysunku 5. Faza gorąca jest potrzebna do tego aby proces wyskakiwał z minimów lokalnych. A podczas fazy chłodzenia sytuacja się coraz bardziej stabilizuje.

Działanie całego procesu

W naszym konkretnym przypadku przyjęto, że na każdą fazę agent grupy, prowadzącego i rocznika, wykonuje pięć ruchów, co oznacza, że w całym cyklu agenci wykonują po 15 ruchów każdy. Ale wyjątkiem są tu agenci roczników, którzy jeszcze po każdym cyklu próbują zastosować jakiś gotowy schemat. Tak więc, agenci roczników wykonują po 16 ruchów na cykl. Do tego jeszcze, w każdym cyklu, agent sal próbuje lepiej uporządkować wykorzystanie sal. Agent centralny koordynuje działania pozostałych agentów. Działanie sekwencyjnego programu wirtualnie symuluje działanie systemu równoległego, dlatego po każdym ruchu agenta, następny ruch wykonuje już inny agent. W przyszłości program wielowątkowy pozwoli rzeczywiście zrównoleglić ten proces, przynajmniej w jakimś stopniu, zależnym od liczby dostępnych niezależnych jednostek obliczeniowych. Ruch agenta może być pusty, jeśli agent nie jest w stanie wykonać ruchu. Na naszym wydziale, obecnie jest 12 roczników, 34 grupy i 76 prowadzących, co po doliczeniu agenta sal i głównego agenta, daje liczbę 124 agentów. W tym przypadku, wszyscy agenci wykonują 1843 ruchy na każdy cykl. Przy czym działania agenta sal zostało tu uwzględnione jako tylko jeden ruch. W tej sytuacji, po wykonaniu stu cykli procesu wszyscy agenci wykonują 184 300 ruchów.

Warunek stopu składa się z trzech warunków, które są razem brane pod uwagę poprzez alternatywę warunków składowych. Pierwszy, to wykonanie pełnych stu cykli procesu optymalizacji, drugi, to brak poprawy wartości globalnej funkcji celu, a trzeci to prośba użytkownika programu o przerwanie procesu. Brak poprawy wartości funkcji celu oznacza dokładnie brak poprawy o 5 punktów podczas ostatnich 10 cykli procesu. Cały proces można przedstawić w postaci schematu blokowego (rys. 6).



Rysunek 6. Schemat blokowy całego procesu optymalizacji opartego na systemie wieloagentowym
 Figure 6. Flowchart of the entire optimization process based on a multi-agent system

Źródło: opracowanie własne

Należy tu jeszcze wyraźnie zaznaczyć, że agenci nie przekładają zajęć innych agentów, tylko własne. Nie mniej jednak, cały plan może dowolnie ewoluować, bo każde przeniesienie zajęć zwalnia jednocześnie salę i czas prowadzącego w tym miejscu planu i w efekcie inne zajęcia można ustalić w tym czasie. Mogą to być zajęcia zupełnie innego rocznika. W efekcie, po wielu zmianach, plan może już wyglądać zupełnie inaczej. Warto zwrócić uwagę na jeszcze inną ważną rzecz. Jeżeli agenci mogą przekładać tylko własne zajęcia, to tak naprawdę bezpośrednio rywalizują tylko agenci grup i roczników z agentami prowadzącymi. Każde zajęcie jest jednocześnie związane z grupą studentów i z pewnym prowadzącym. Jeżeli preferencje prowadzącego stoją w konflikcie z preferencjami studentów, to w fazie gorącej i w fazie chłodzenia może dochodzić do działań przeciwstawnych. Agenci grup i roczników rywalizują między sobą już tylko pośrednio. Przełożenie zajęć jednej grupy może uniemożliwić przełożenie zajęć innej grupy w to samo miejsce planu, z powodu możliwych kolizji sal i prowadzących. Na tej zasadzie mogą rywalizować ze sobą także agenci dwóch grup, dwóch roczników, a także dwóch prowadzących. Oczywiście najczęściej takich sytuacji jest w fazie gorącej każdego cyklu procesu. Rozważane jest także dodanie czwartej fazy, super gorącej, podczas której agenci będą mogli także przenosić zajęcia innych agentów oraz wykonywać zmiany losowe. Wbrew pozorom, takie rozwiązanie będzie znacznie bardziej złożone algorytmicznie. Przenoszenie losowe zajęć, w fazie super gorącej, mogło by względnie szybko doprowadzić do zupełnie nowych konfiguracji planu zajęć i tym samym bardzo silnie wybijać ułożenia planów z minimów lokalnych.

Podsumowanie i wnioski

Układanie planu zajęć na wyższej uczelni jest bardzo ważnym i trudnym zadaniem. Ogromne znaczenie mają więc próby tworzenia narzędzi do automatycznej optymalizacji planu, jednak ten problem jest niezwykle złożonym zagadnieniem. Obecnie opracowano i stworzono bardzo wygodne narzędzie do układania planu, głównie dedykowane do układania planów na wydziale zastosowań informatyki i matematyki SGGW. Na tę chwilę, plan jest jeszcze tworzony przez planistę, który jest użytkownikiem programu komputerowego. W dalszym ciągu prac planowana jest implementacja heurystycznych algorytmów do automatycznego wspomaganie układania planu, w tym także dla studiów zaocznych. System wspomagający jest już w trakcie tworzenia i przeprowadzono już pierwsze próby jego działania. Proponowany system jest głównie oparty na zastosowaniu systemu wieloagentowego. Wyniki działania są bardzo obiecujące. Jednak, na tę chwilę bardzo trudno jeszcze ściśle ocenić efekty działania, ponieważ wymierna ocena wydajności jest jeszcze nie zdefiniowana. Badane narzędzie ma wspomagać pracę planisty, tak więc, bardzo dużo zależy od tego, jak sprawny jest ten planista układający plany ręcznie. Jeśli planista jest początkujący i bardzo niewprawnie układa plany, to takie narzędzie automatycznego wspomaganie może istotnie poprawić plan przez niego ułożony, ale jeśli planista jest doświadczony i bardzo sprawnie układa plany, to może się nawet zdarzyć, że takie narzędzie popsuje efekty jego pracy. Do tego jeszcze dochodzi jeden ważny aspekt. Proponowane narzędzie jest głównie projektowane pod kątem tworzenia całkowicie nowych planów, jako narzędzie wspomagające proces układania. Jeśli plany zajęć są już od wielu lat ustabilizowane i wszyscy są do tych planów przyzwyczajeni, to bardzo trudno coś rzeczywiście poprawić. Taka sytuacja może mieć miejsce, gdy na przykład struktura zajęć jest bardzo niezmienna, stale są te same przedmioty, podobne liczby grup studenckich i specjalizacje, ci sami prowadzący. Wówczas stosowanie takich narzędzi może być tylko dość ryzykowne. Złożoność obliczeniowa tworzenia i poprawiania planu zajęć jest tak wielka, że naprawdę skuteczne działanie jest niezwykle trudne, więc trudno istotnie poprawić plan, który jest już uznawany za dobry. Drugą istotną rzeczą jest fakt, że działanie tego algorytmu jest silnie uzależnione od liczby stopni swobody podczas procesu układania w istniejącym planie zajęć. Jeśli liczba zajęć do ułożenia jest względnie duża względem dostępnych zasobów (czyli liczba sal, prowadzących i czasu do wykorzystania) to jest bardzo mała swoboda ruchu. W proponowanym algorytmie, każdy agent wykonuje w jednym kroku jeden ruch i ma jednocześnie to być ruch zawsze korzystny. Jednocześnie agent nie przesuwają zajęć innego agenta. Powoduje to, że przy małej swobodzie ruchu, bardzo często nie jest w stanie wykonać jakiegokolwiek ruchu. Stąd efektywność działania algorytmu zależy od stopni swobody. Tak więc, w pewnych warunkach algorytm może być bardzo przydatny, a w innych nie. Na przykład, gdy cały plan ma dużą złożoność, ale jednocześnie i dużo stopni swobody dzięki temu, że dostępne zasoby są odpowiednio duże, wówczas układanie planu będzie bardzo złożone dla planisty, a program komputerowy może z tym sobie całkiem dobrze poradzić. W przypadku dużych ograniczeń swobody ruchu, w wielu sytuacjach, pary agentów lub większe ich grupy powinny wspólnie razem ustalać swoje ruchy. Jednak, wbrew pozorom samo szukanie lepszego rozwiązania nie będzie tu największym problemem. Jeszcze trudniej-

szym zadaniem będzie rozpoznawanie takich sytuacji w planie, oraz ustalanie agentów, którzy powinni brać aktywny udział w tym procesie lokalnej optymalizacji części planu. Do poszukiwania lepszych lokalnych rozwiązań można, na przykład, użyć algorytmów ewolucyjnych [n]. Rozważane są także próby zastosowania algorytmów grawitacyjnych, algorytmu świetlika, inteligencji roju lub automatów komórkowych. To właśnie będzie przedmiotem dalszych badań w tym kierunku. Jeszcze innym kierunkiem rozwoju może być tworzenie systemu wieloagentowego, gdzie rozwiązywanie konfliktów będzie się odbywało na drodze negocjacji między agentami. Niezależnie od tego, autorzy są głęboko przekonani, że zastosowanie systemów wieloagentowych do automatycznego układania planu zajęć jest bardzo trafnym podejściem, ale swoją drogą, taki system powinien być również wzmocniony przez zastosowanie innych technik sztucznej inteligencji.

Bibliografia

- Ahmad I.R., Sufahani S., Ali M., Razali S.N., 2018: A heuristics approach for classroom scheduling using genetic algorithm technique, *Journal of Physics: Conference Series*, 995, 1, 012050.
- Chan C.K., Gooi H.B., Lim M.H., 2002: Co-evolutionary algorithm approach to a university timetable system, *Proceedings of the 2002 Congress on Evolutionary Computation, CEC'02* (Cat. No. 02TH8600), 2, 195–199.
- Corman F., Meng L., 2014: A review of online dynamic models and algorithms for railway traffic management, *IEEE Transactions on Intelligent Transportation Systems*, 16(3), 1274–1284.
- Deris S., Omatu S., Ohta H., Saad P., 1999: Incorporating constraint propagation in genetic algorithm for university timetable planning. *Engineering applications of artificial intelligence*, 12(3), 241–253.
- Deris S., Omatu S., Ohta H., 2000: Timetable planning using the constraint-based reasoning, *Computers & Operations Research*, 27(9), 819–840.
- Flasiński M., 2011: *Wstęp do sztucznej inteligencji*, Wydawnictwo PWN, Warszawa.
- Goosuerw J.W.H.M., 2004: *Models and algorithms for railway line planning problems*, Universiteit Maastricht, Maastricht.
- Huhns M.N., Stephens L.M., 1999: Multiagent systems and societies of agents. *Multiagent systems: a modern approach to distributed artificial intelligence*, 1, 79–114.
- Irene H.S.F., Deris S., Hashim S.Z.M., 2009: University course timetable planning using hybrid particle swarm optimization. *Proceedings of the first ACM/SIGEVO Summit on Genetic and Evolutionary Computation*, 239–246.
- Liebchen C., 2008: The first optimized railway timetable in practice, *Transportation Science*, 42(4), 420–435.
- Mittal D., Doshi H., Sunasra M., Nagpure R., 2015: Automatic timetable generation using genetic algorithm, *International Journal of Advanced Research in Computer and Communication Engineering*, 4(2), 245–248.
- Papadimitriou C.H., Płoski Z., 2012: *Złożoność obliczeniowa*, Helion, Gliwice.
- Prosad R., Khan M.A.R., Ahammad I., 2022: Design of Class Routine and Exam Hall Invigilation System based on Genetic Algorithm and Greedy Approach, *Asian Journal of Research in Computer Science*, 13(3), 28–44.

- Rutkowski L., 2006: *Metody i techniki sztucznej inteligencji: inteligencja obliczeniowa*, Wydawnictwo Naukowe PWN, Warszawa.
- Wagner D., 2003: *Algorithms and models for railway optimization*, Workshop on Algorithms and Data Structures, Springer, Berlin, Heidelberg.
- Weiss G. (red.), 1999: *Multiagent systems: a modern approach to distributed artificial intelligence*, MIT press, Cambridge, Massachusetts.